

Six Degrees of Kevin Bacon

Introduction - Six Degrees of Kevin Bacon is a game based on the "six degrees of separation" concept, which posits that any two people on Earth are six or fewer acquaintance links apart. Movie buffs challenge each other to find the shortest path between an arbitrary actor and prolific actor Kevin Bacon. It rests on the assumption that anyone involved in the film industry can be linked through their film roles to Bacon within six steps. The analysis of social networks can be a computationally intensive task, especially when dealing with large volumes of data. It is also a challenging problem to devise a correct methodology to infer an informative social network structure. Here, we will analyze a social network of actors and actresses that co-participated in movies. We will do some simple descriptive analysis, and in the end try to relate an actor/actress's position in the social network with the success of the movies in which they participate.

Rules & Notes - Please take your time to read the following points:

1. The submission deadline will be set for the 30th of May at 23:59h.
2. It is acceptable that you **discuss** with your colleagues different approaches to solve each step of the problem set, but the assignment is individual. That is, you are responsible for writing your own code, and analyzing the results. Clear cases of cheating will be penalized with 0 points in this assignment.
3. After review of your submission files, and before a mark is attributed, you might be called to orally defend your submission.
4. You will be scored first and foremost by the number of correct answers, secondly by the logic used in trying to approach each step of the problem set.
5. You can add as many cells as you like to answer the questions.
6. It is also important you clearly indicate what your final answer to each question is when you are using multiple cells (for example you can use `print("My final answer is:")` before your answer or use cell comments).
7. Consider skipping questions that you are stuck on, and get back to them later.
8. Expect computations to take a few minutes to finish in some of the steps.
9. It is recommended you read the whole assignment before starting.
10. You can make use of caching or persisting your RDDs or Dataframes, this may speed up performance. You do not need to cache every dataframe, but usually you want to do this at least once after the data has been imported.
11. If you have trouble with graphframes in databricks (specifically the import statement) you need to make sure the graphframes package is installed on the cluster you are running. If you click home on the left, then click on the graphframes library which you loaded in Lab 11 you can install the package on your cluster (check the graphframes checkbox and click install)
12. Be careful, you must not 'Publish' this notebook in databricks.
13. **IMPORTANT** It is expected you have developed skills beyond writing SQL queries. Any question where you directly write a SQL query (by for example creating a temporary view and then using `spark.sql` to pass the query) will receive a 25% penalty. Using the spark syntax (for example `dataframe.select(""").where("conditions")`) is acceptable and does not incur this penalty.
14. **Questions** – Any questions about this assignment should be posted in the Forum@Moodle. Questions by e-mail will not be answered. The lab will run at the normal time. During this period you can ask any questions you have about the exam (we can't provide you the actual answers of course, but there may be helpful tips if you are stuck on any of the steps). As such, it is probably useful to attempt the assignment before the scheduled lab.
15. **Delivery** - To fulfil this activity you will have to upload the following materials to Moodle:

- An exported IPython notebook. From the menu at the top, select 'File', then 'Export', then 'IPython Notebook', to download the notebook. The notebook should be solved (have results displayed), but should contain all necessary code so that when the notebook is run in databricks it should also replicate these results. This means that all data downloading and processing should be done in this notebook. It is also important you clearly indicate where your final answer to each question is when you are using multiple cells (for example you can use `print("My final answer is:")` before your answer or use cell comments).
- A PDF version of your code and answers. There are a couple of ways you can do this. You can convert the downloaded IPython Notebook to pdf (check out nbconvert if you have Jupyter notebook), or you can just copy your code and answers into a word file and save as pdf, or finally you can take screenshots of each page of the notebook and put them into a word file and save it as pdf. It is important that all code and answers are visible in this pdf.
- You will also need to provide a signed statement of authorship, which is available on Moodle.

Data Sources and Description

We will use data from IMDB. You can download raw datafiles from <https://datasets.imdbws.com> (<https://datasets.imdbws.com>). Note that the files are tab delimited (.tsv) You can find a description of the each datafile in <https://www.imdb.com/interfaces/> (<https://www.imdb.com/interfaces/>)

Questions

Data loading and preparation

Review the file descriptions and load the necessary data onto your databricks cluster and into spark dataframes or rdds. You will need to use shell commands to download the data, unzip the data, load the data into spark. Note that the data might require parsing and preprocessing to be ready for the questions below.

Hints You can use `gunzip` to unzip the .gz files. The data files will then be tab separated (.tsv), which you can load into a dataframe using the tab separated option instead of the comma separated option we have typically used in class: `.option("sep", "\t")`

```
In [3]: %sh wget https://datasets.imdbws.com/name.basics.tsv.gz  
%sh  
gunzip name.basics.tsv.gz
```

```
--2020-05-31 14:07:15-- https://datasets.imdbws.com/name.basics.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.37, 13.224.13.54,
13.224.13.26, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.37|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 197672123 (189M) [binary/octet-stream]
Saving to: 'name.basics.tsv.gz'
```

0K	0%	7.24M	26s
50K	0%	6.69M	27s
100K	0%	14.2M	22s
150K	0%	12.4M	21s
200K	0%	21.2M	18s
250K	0%	31.0M	16s
300K	0%	23.8M	15s
350K	0%	31.6M	14s
400K	0%	50.9M	13s
450K	0%	33.3M	12s
500K	0%	60.0M	11s
550K	0%	54.1M	11s
600K	0%	48.2M	10s
650K	0%	49.4M	10s
700K	0%	53.4M	9s
750K	0%	84.3M	9s
800K	0%	79.3M	8s
850K	0%	11.9M	9s
900K	0%	84.9M	8s
950K	0%	138M	8s
1000K	0%	126M	8s
1050K	0%	110M	8s
1100K	0%	115M	7s
1150K	0%	113M	7s
1200K	0%	173M	7s
1250K	0%	176M	7s
1300K	0%	209M	6s
1350K	0%	186M	6s
1400K	0%	279M	6s
1450K	0%	177M	6s
1500K	0%	226M	6s
1550K	0%	222M	5s
1600K	0%	224M	5s
1650K	0%	215M	5s
1700K	0%	204M	5s
1750K	0%	273M	5s
1800K	0%	193M	5s
1850K	0%	166M	5s
1900K	1%	223M	5s
1950K	1%	230M	5s
2000K	1%	287M	4s
2050K	1%	239M	4s

2100K	1%	65.0M	4s
2150K	1%	152M	4s
2200K	1%	152M	4s
2250K	1%	107M	4s
2300K	1%	185M	4s
2350K	1%	188M	4s
2400K	1%	170M	4s
2450K	1%	144M	4s
2500K	1%	192M	4s
2550K	1%	172M	4s
2600K	1%	205M	4s
2650K	1%	175M	4s
2700K	1%	275M	4s
2750K	1%	297M	4s
2800K	1%	177M	4s
2850K	1%	155M	3s
2900K	1%	242M	3s
2950K	1%	191M	3s
3000K	1%	257M	3s
3050K	1%	156M	3s
3100K	1%	269M	3s
3150K	1%	296M	3s
3200K	1%	173M	3s
3250K	1%	231M	3s
3300K	1%	178M	3s
3350K	1%	223M	3s
3400K	1%	289M	3s
3450K	1%	211M	3s
3500K	1%	258M	3s
3550K	1%	290M	3s
3600K	1%	273M	3s
3650K	1%	48.2M	3s
3700K	1%	97.3M	3s
3750K	1%	143M	3s
3800K	1%	89.2M	3s
3850K	2%	111M	3s
3900K	2%	135M	3s
3950K	2%	127M	3s
4000K	2%	156M	3s
4050K	2%	158M	3s
4100K	2%	163M	3s
4150K	2%	80.4M	3s
4200K	2%	174M	3s
4250K	2%	119M	3s
4300K	2%	135M	3s
4350K	2%	62.7M	3s
4400K	2%	114M	3s
4450K	2%	127M	3s
4500K	2%	150M	3s
4550K	2%	134M	3s
4600K	2%	112M	3s

4650K	2%	118M	3s
4700K	2%	186M	3s
4750K	2%	115M	3s
4800K	2%	162M	3s
4850K	2%	129M	3s
4900K	2%	140M	3s
4950K	2%	140M	3s
5000K	2%	124M	3s
5050K	2%	149M	3s
5100K	2%	170M	3s
5150K	2%	158M	2s
5200K	2%	119M	2s
5250K	2%	136M	2s
5300K	2%	132M	2s
5350K	2%	97.8M	2s
5400K	2%	134M	2s
5450K	2%	119M	2s
5500K	2%	155M	2s
5550K	2%	131M	2s
5600K	2%	125M	2s
5650K	2%	104M	2s
5700K	2%	129M	2s
5750K	3%	86.4M	2s
5800K	3%	120M	2s
5850K	3%	133M	2s
5900K	3%	123M	2s
5950K	3%	76.7M	2s
6000K	3%	143M	2s
6050K	3%	130M	2s
6100K	3%	105M	2s
6150K	3%	111M	2s
6200K	3%	147M	2s
6250K	3%	99.7M	2s
6300K	3%	143M	2s
6350K	3%	144M	2s
6400K	3%	129M	2s
6450K	3%	145M	2s
6500K	3%	136M	2s
6550K	3%	150M	2s
6600K	3%	102M	2s
6650K	3%	120M	2s
6700K	3%	106M	2s
6750K	3%	146M	2s
6800K	3%	141M	2s
6850K	3%	113M	2s
6900K	3%	49.0M	2s
6950K	3%	38.8M	2s
7000K	3%	40.4M	2s
7050K	3%	40.1M	2s
7100K	3%	36.4M	2s
7150K	3%	62.3M	2s

7200K	3%	146M	2s
7250K	3%	85.0M	2s
7300K	3%	118M	2s
7350K	3%	132M	2s
7400K	3%	141M	2s
7450K	3%	116M	2s
7500K	3%	131M	2s
7550K	3%	145M	2s
7600K	3%	134M	2s
7650K	3%	128M	2s
7700K	4%	126M	2s
7750K	4%	147M	2s
7800K	4%	125M	2s
7850K	4%	120M	2s
7900K	4%	149M	2s
7950K	4%	128M	2s
8000K	4%	155M	2s
8050K	4%	136M	2s
8100K	4%	128M	2s
8150K	4%	149M	2s
8200K	4%	144M	2s
8250K	4%	107M	2s
8300K	4%	141M	2s
8350K	4%	138M	2s
8400K	4%	149M	2s
8450K	4%	145M	2s
8500K	4%	123M	2s
8550K	4%	139M	2s
8600K	4%	132M	2s
8650K	4%	101M	2s
8700K	4%	94.2M	2s
8750K	4%	134M	2s
8800K	4%	52.9M	2s
8850K	4%	31.0M	2s
8900K	4%	36.9M	2s
8950K	4%	36.6M	2s
9000K	4%	37.4M	2s
9050K	4%	31.4M	2s
9100K	4%	22.6M	2s
9150K	4%	35.1M	2s
9200K	4%	37.8M	2s
9250K	4%	37.3M	2s
9300K	4%	76.9M	2s
9350K	4%	125M	2s
9400K	4%	146M	2s
9450K	4%	113M	2s
9500K	4%	158M	2s
9550K	4%	160M	2s
9600K	4%	162M	2s
9650K	5%	113M	2s
9700K	5%	146M	2s

9750K	5%	149M	2s
9800K	5%	189M	2s
9850K	5%	97.9M	2s
9900K	5%	169M	2s
9950K	5%	139M	2s
10000K	5%	163M	2s
10050K	5%	149M	2s
10100K	5%	140M	2s
10150K	5%	118M	2s
10200K	5%	146M	2s
10250K	5%	122M	2s
10300K	5%	135M	2s
10350K	5%	161M	2s
10400K	5%	159M	2s
10450K	5%	139M	2s
10500K	5%	137M	2s
10550K	5%	145M	2s
10600K	5%	149M	2s
10650K	5%	123M	2s
10700K	5%	160M	2s
10750K	5%	160M	2s
10800K	5%	163M	2s
10850K	5%	148M	2s
10900K	5%	121M	2s
10950K	5%	81.3M	2s
11000K	5%	117M	2s
11050K	5%	138M	2s
11100K	5%	93.1M	2s
11150K	5%	168M	2s
11200K	5%	115M	2s
11250K	5%	144M	2s
11300K	5%	143M	2s
11350K	5%	139M	2s
11400K	5%	143M	2s
11450K	5%	103M	2s
11500K	5%	240M	2s
11550K	6%	254M	2s
11600K	6%	289M	2s
11650K	6%	125M	2s
11700K	6%	148M	2s
11750K	6%	143M	2s
11800K	6%	151M	2s
11850K	6%	128M	2s
11900K	6%	143M	2s
11950K	6%	244M	2s
12000K	6%	183M	2s
12050K	6%	169M	2s
12100K	6%	193M	2s
12150K	6%	194M	2s
12200K	6%	205M	2s
12250K	6%	165M	2s

12300K	6%	147M	2s
12350K	6%	182M	2s
12400K	6%	135M	2s
12450K	6%	115M	2s
12500K	6%	141M	2s
12550K	6%	145M	2s
12600K	6%	222M	2s
12650K	6%	129M	2s
12700K	6%	232M	2s
12750K	6%	192M	2s
12800K	6%	178M	2s
12850K	6%	191M	2s
12900K	6%	188M	2s
12950K	6%	165M	2s
13000K	6%	174M	2s
13050K	6%	108M	2s
13100K	6%	151M	2s
13150K	6%	149M	2s
13200K	6%	147M	2s
13250K	6%	128M	2s
13300K	6%	165M	2s
13350K	6%	69.7M	2s
13400K	6%	76.1M	2s
13450K	6%	96.1M	2s
13500K	7%	116M	2s
13550K	7%	98.8M	2s
13600K	7%	105M	2s
13650K	7%	95.1M	2s
13700K	7%	131M	2s
13750K	7%	80.8M	2s
13800K	7%	93.3M	2s
13850K	7%	97.2M	2s
13900K	7%	155M	2s
13950K	7%	150M	2s
14000K	7%	134M	2s
14050K	7%	137M	2s
14100K	7%	114M	2s
14150K	7%	69.6M	2s
14200K	7%	93.4M	2s
14250K	7%	124M	2s
14300K	7%	120M	2s
14350K	7%	153M	2s
14400K	7%	115M	2s
14450K	7%	139M	2s
14500K	7%	145M	2s
14550K	7%	125M	2s
14600K	7%	141M	2s
14650K	7%	131M	2s
14700K	7%	142M	2s
14750K	7%	152M	2s
14800K	7%	141M	2s

14850K	7%	115M	2s
14900K	7%	139M	2s
14950K	7%	133M	2s
15000K	7%	148M	2s
15050K	7%	121M	2s
15100K	7%	159M	2s
15150K	7%	120M	2s
15200K	7%	87.6M	2s
15250K	7%	138M	2s
15300K	7%	111M	2s
15350K	7%	138M	2s
15400K	8%	96.6M	2s
15450K	8%	122M	2s
15500K	8%	149M	2s
15550K	8%	138M	2s
15600K	8%	99.0M	2s
15650K	8%	126M	2s
15700K	8%	141M	2s
15750K	8%	125M	2s
15800K	8%	140M	2s
15850K	8%	125M	2s
15900K	8%	122M	2s
15950K	8%	196M	2s
16000K	8%	136M	2s
16050K	8%	124M	2s
16100K	8%	116M	2s
16150K	8%	124M	2s

*** WARNING: skipped 244036 bytes of output ***

176750K	91%	146M	0s
176800K	91%	208M	0s
176850K	91%	205M	0s
176900K	91%	193M	0s
176950K	91%	169M	0s
177000K	91%	242M	0s
177050K	91%	96.7M	0s
177100K	91%	130M	0s
177150K	91%	146M	0s
177200K	91%	119M	0s
177250K	91%	141M	0s
177300K	91%	165M	0s
177350K	91%	185M	0s
177400K	91%	178M	0s
177450K	91%	170M	0s
177500K	91%	233M	0s
177550K	92%	244M	0s
177600K	92%	291M	0s
177650K	92%	291M	0s
177700K	92%	267M	0s
177750K	92%	111M	0s

177800K	92%	82.4M	0s
177850K	92%	132M	0s
177900K	92%	117M	0s
177950K	92%	152M	0s
178000K	92%	149M	0s
178050K	92%	148M	0s
178100K	92%	146M	0s
178150K	92%	132M	0s
178200K	92%	149M	0s
178250K	92%	202M	0s
178300K	92%	109M	0s
178350K	92%	266M	0s
178400K	92%	167M	0s
178450K	92%	167M	0s
178500K	92%	180M	0s
178550K	92%	182M	0s
178600K	92%	167M	0s
178650K	92%	167M	0s
178700K	92%	231M	0s
178750K	92%	168M	0s
178800K	92%	245M	0s
178850K	92%	289M	0s
178900K	92%	225M	0s
178950K	92%	296M	0s
179000K	92%	197M	0s
179050K	92%	137M	0s
179100K	92%	111M	0s
179150K	92%	117M	0s
179200K	92%	137M	0s
179250K	92%	173M	0s
179300K	92%	160M	0s
179350K	92%	173M	0s
179400K	92%	138M	0s
179450K	92%	153M	0s
179500K	93%	138M	0s
179550K	93%	204M	0s
179600K	93%	189M	0s
179650K	93%	171M	0s
179700K	93%	125M	0s
179750K	93%	187M	0s
179800K	93%	155M	0s
179850K	93%	169M	0s
179900K	93%	138M	0s
179950K	93%	152M	0s
180000K	93%	162M	0s
180050K	93%	158M	0s
180100K	93%	158M	0s
180150K	93%	182M	0s
180200K	93%	181M	0s
180250K	93%	161M	0s
180300K	93%	122M	0s

180350K	93%	155M	0s
180400K	93%	143M	0s
180450K	93%	110M	0s
180500K	93%	127M	0s
180550K	93%	116M	0s
180600K	93%	103M	0s
180650K	93%	98.1M	0s
180700K	93%	89.4M	0s
180750K	93%	72.8M	0s
180800K	93%	118M	0s
180850K	93%	123M	0s
180900K	93%	104M	0s
180950K	93%	169M	0s
181000K	93%	108M	0s
181050K	93%	154M	0s
181100K	93%	157M	0s
181150K	93%	185M	0s
181200K	93%	157M	0s
181250K	93%	156M	0s
181300K	93%	145M	0s
181350K	93%	187M	0s
181400K	93%	160M	0s
181450K	94%	189M	0s
181500K	94%	155M	0s
181550K	94%	188M	0s
181600K	94%	191M	0s
181650K	94%	194M	0s
181700K	94%	167M	0s
181750K	94%	171M	0s
181800K	94%	191M	0s
181850K	94%	193M	0s
181900K	94%	161M	0s
181950K	94%	189M	0s
182000K	94%	168M	0s
182050K	94%	154M	0s
182100K	94%	161M	0s
182150K	94%	203M	0s
182200K	94%	281M	0s
182250K	94%	241M	0s
182300K	94%	164M	0s
182350K	94%	282M	0s
182400K	94%	256M	0s
182450K	94%	290M	0s
182500K	94%	270M	0s
182550K	94%	267M	0s
182600K	94%	270M	0s
182650K	94%	283M	0s
182700K	94%	142M	0s
182750K	94%	182M	0s
182800K	94%	185M	0s
182850K	94%	294M	0s

182900K	94%	63.6M	0s
182950K	94%	137M	0s
183000K	94%	152M	0s
183050K	94%	142M	0s
183100K	94%	127M	0s
183150K	94%	152M	0s
183200K	94%	107M	0s
183250K	94%	180M	0s
183300K	94%	116M	0s
183350K	95%	64.9M	0s
183400K	95%	137M	0s
183450K	95%	126M	0s
183500K	95%	99.6M	0s
183550K	95%	136M	0s
183600K	95%	122M	0s
183650K	95%	103M	0s
183700K	95%	100M	0s
183750K	95%	51.5M	0s
183800K	95%	132M	0s
183850K	95%	149M	0s
183900K	95%	124M	0s
183950K	95%	160M	0s
184000K	95%	113M	0s
184050K	95%	73.3M	0s
184100K	95%	108M	0s
184150K	95%	148M	0s
184200K	95%	132M	0s
184250K	95%	73.3M	0s
184300K	95%	101M	0s
184350K	95%	148M	0s
184400K	95%	155M	0s
184450K	95%	149M	0s
184500K	95%	144M	0s
184550K	95%	148M	0s
184600K	95%	147M	0s
184650K	95%	152M	0s
184700K	95%	128M	0s
184750K	95%	145M	0s
184800K	95%	153M	0s
184850K	95%	159M	0s
184900K	95%	138M	0s
184950K	95%	160M	0s
185000K	95%	109M	0s
185050K	95%	126M	0s
185100K	95%	102M	0s
185150K	95%	126M	0s
185200K	95%	103M	0s
185250K	95%	169M	0s
185300K	96%	96.0M	0s
185350K	96%	143M	0s
185400K	96%	135M	0s

185450K	96%	156M	0s
185500K	96%	131M	0s
185550K	96%	158M	0s
185600K	96%	154M	0s
185650K	96%	114M	0s
185700K	96%	99.0M	0s
185750K	96%	139M	0s
185800K	96%	155M	0s
185850K	96%	134M	0s
185900K	96%	114M	0s
185950K	96%	99.0M	0s
186000K	96%	69.8M	0s
186050K	96%	83.6M	0s
186100K	96%	67.3M	0s
186150K	96%	89.8M	0s
186200K	96%	104M	0s
186250K	96%	175M	0s
186300K	96%	148M	0s
186350K	96%	153M	0s
186400K	96%	182M	0s
186450K	96%	181M	0s
186500K	96%	59.6M	0s
186550K	96%	93.3M	0s
186600K	96%	115M	0s
186650K	96%	176M	0s
186700K	96%	141M	0s
186750K	96%	175M	0s
186800K	96%	87.3M	0s
186850K	96%	71.6M	0s
186900K	96%	113M	0s
186950K	96%	183M	0s
187000K	96%	39.8M	0s
187050K	96%	172M	0s
187100K	96%	136M	0s
187150K	96%	183M	0s
187200K	97%	179M	0s
187250K	97%	183M	0s
187300K	97%	254M	0s
187350K	97%	289M	0s
187400K	97%	128M	0s
187450K	97%	172M	0s
187500K	97%	111M	0s
187550K	97%	171M	0s
187600K	97%	158M	0s
187650K	97%	204M	0s
187700K	97%	261M	0s
187750K	97%	266M	0s
187800K	97%	295M	0s
187850K	97%	291M	0s
187900K	97%	88.0M	0s
187950K	97%	112M	0s

188000K	97%	189M	0s
188050K	97%	112M	0s
188100K	97%	151M	0s
188150K	97%	179M	0s
188200K	97%	173M	0s
188250K	97%	179M	0s
188300K	97%	144M	0s
188350K	97%	179M	0s
188400K	97%	174M	0s
188450K	97%	191M	0s
188500K	97%	145M	0s
188550K	97%	175M	0s
188600K	97%	177M	0s
188650K	97%	101M	0s
188700K	97%	87.1M	0s
188750K	97%	126M	0s
188800K	97%	172M	0s
188850K	97%	158M	0s
188900K	97%	115M	0s
188950K	97%	158M	0s
189000K	97%	182M	0s
189050K	97%	98.4M	0s
189100K	97%	80.5M	0s
189150K	98%	72.9M	0s
189200K	98%	95.3M	0s
189250K	98%	170M	0s
189300K	98%	164M	0s
189350K	98%	174M	0s
189400K	98%	183M	0s
189450K	98%	192M	0s
189500K	98%	144M	0s
189550K	98%	169M	0s
189600K	98%	174M	0s
189650K	98%	184M	0s
189700K	98%	80.6M	0s
189750K	98%	177M	0s
189800K	98%	154M	0s
189850K	98%	180M	0s
189900K	98%	118M	0s
189950K	98%	153M	0s
190000K	98%	171M	0s
190050K	98%	121M	0s
190100K	98%	160M	0s
190150K	98%	39.5M	0s
190200K	98%	44.5M	0s
190250K	98%	45.1M	0s
190300K	98%	31.5M	0s
190350K	98%	55.0M	0s
190400K	98%	100M	0s
190450K	98%	89.1M	0s
190500K	98%	84.5M	0s

190550K	98%	95.5M	0s
190600K	98%	109M	0s
190650K	98%	108M	0s
190700K	98%	149M	0s
190750K	98%	180M	0s
190800K	98%	165M	0s
190850K	98%	182M	0s
190900K	98%	157M	0s
190950K	98%	183M	0s
191000K	98%	177M	0s
191050K	98%	158M	0s
191100K	99%	133M	0s
191150K	99%	179M	0s
191200K	99%	97.3M	0s
191250K	99%	108M	0s
191300K	99%	159M	0s
191350K	99%	180M	0s
191400K	99%	183M	0s
191450K	99%	182M	0s
191500K	99%	146M	0s
191550K	99%	182M	0s
191600K	99%	184M	0s
191650K	99%	182M	0s
191700K	99%	166M	0s
191750K	99%	180M	0s
191800K	99%	184M	0s
191850K	99%	166M	0s
191900K	99%	119M	0s
191950K	99%	176M	0s
192000K	99%	156M	0s
192050K	99%	104M	0s
192100K	99%	92.9M	0s
192150K	99%	152M	0s
192200K	99%	170M	0s
192250K	99%	175M	0s
192300K	99%	152M	0s
192350K	99%	182M	0s
192400K	99%	186M	0s
192450K	99%	167M	0s
192500K	99%	161M	0s
192550K	99%	187M	0s
192600K	99%	173M	0s
192650K	99%	182M	0s
192700K	99%	148M	0s
192750K	99%	183M	0s
192800K	99%	185M	0s
192850K	99%	149M	0s
192900K	99%	166M	0s
192950K	99%	181M	0s
193000K	100%	159M=1.9s	

2020-05-31 14:07:17 (98.0 MB/s) - 'name.basics.tsv.gz' saved [197672123/197672123]

/bin/bash: line 1: fg: no job control

gzip: name.basics.tsv already exists; not overwritten

```
In [4]: names_basics = spark.read.option("sep", "\t").csv('file:/databricks/driver/name.basics.tsv', header=True, inferSchema = True)
names_basics.cache()
names_basics.show(3)
```

```
+-----+-----+-----+-----+-----+-----+
+-----+
nconst|    primaryName|birthYear|deathYear|    primaryProfession|    knownForTitles|
+-----+-----+-----+-----+-----+-----+
+-----+
nm0000001|    Fred Astaire|    1899|    1987|soundtrack,actor,...|tt0053137,tt00504...|
nm0000002|    Lauren Bacall|    1924|    2014|    actress,soundtrack|tt0117057,tt00718...|
nm0000003|Brigitte Bardot|    1934|    \N|actress,soundtrac...|tt0054452,tt00599...|
+-----+-----+-----+-----+-----+-----+
+-----+
only showing top 3 rows
```

```
In [5]: %sh wget https://datasets.imdbws.com/title.akas.tsv.gz  
%sh  
gunzip title.akas.tsv.gz
```

```
--2020-05-31 14:07:46-- https://datasets.imdbws.com/title.akas.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.32, 13.224.13.37,
13.224.13.54, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.32|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 192780293 (184M) [binary/octet-stream]
Saving to: 'title.akas.tsv.gz'
```

0K	0%	4.57M	40s
50K	0%	8.24M	31s
100K	0%	13.3M	25s
150K	0%	16.0M	22s
200K	0%	3.53M	28s
250K	0%	37.9M	24s
300K	0%	38.2M	21s
350K	0%	35.0M	19s
400K	0%	40.0M	18s
450K	0%	14.9M	17s
500K	0%	34.2M	16s
550K	0%	32.4M	15s
600K	0%	37.5M	14s
650K	0%	37.2M	14s
700K	0%	15.4M	14s
750K	0%	31.3M	13s
800K	0%	37.8M	13s
850K	0%	33.4M	12s
900K	0%	34.7M	12s
950K	0%	31.9M	12s
1000K	0%	35.7M	11s
1050K	0%	36.0M	11s
1100K	0%	33.6M	11s
1150K	0%	31.2M	10s
1200K	0%	36.7M	10s
1250K	0%	34.3M	10s
1300K	0%	35.0M	10s
1350K	0%	33.7M	10s
1400K	0%	39.1M	10s
1450K	0%	37.1M	9s
1500K	0%	36.8M	9s
1550K	0%	33.5M	9s
1600K	0%	37.8M	9s
1650K	0%	37.6M	9s
1700K	0%	38.0M	9s
1750K	0%	31.0M	9s
1800K	0%	37.7M	9s
1850K	1%	37.8M	8s
1900K	1%	39.5M	8s
1950K	1%	31.4M	8s
2000K	1%	37.0M	8s
2050K	1%	37.1M	8s

2100K	1%	38.6M	8s
2150K	1%	33.2M	8s
2200K	1%	33.9M	8s
2250K	1%	33.1M	8s
2300K	1%	34.1M	8s
2350K	1%	29.8M	8s
2400K	1%	34.3M	8s
2450K	1%	35.9M	8s
2500K	1%	37.8M	8s
2550K	1%	32.6M	8s
2600K	1%	49.4M	7s
2650K	1%	33.7M	7s
2700K	1%	156M	7s
2750K	1%	101M	7s
2800K	1%	38.4M	7s
2850K	1%	35.0M	7s
2900K	1%	39.9M	7s
2950K	1%	34.3M	7s
3000K	1%	36.2M	7s
3050K	1%	80.2M	7s
3100K	1%	201M	7s
3150K	1%	161M	7s
3200K	1%	46.5M	7s
3250K	1%	33.7M	7s
3300K	1%	35.6M	7s
3350K	1%	32.3M	7s
3400K	1%	36.4M	7s
3450K	1%	33.3M	7s
3500K	1%	38.8M	7s
3550K	1%	31.9M	7s
3600K	1%	38.8M	7s
3650K	1%	104M	6s
3700K	1%	33.8M	6s
3750K	2%	43.4M	6s
3800K	2%	176M	6s
3850K	2%	195M	6s
3900K	2%	42.5M	6s
3950K	2%	28.9M	6s
4000K	2%	34.3M	6s
4050K	2%	33.4M	6s
4100K	2%	34.6M	6s
4150K	2%	31.6M	6s
4200K	2%	49.0M	6s
4250K	2%	37.6M	6s
4300K	2%	51.4M	6s
4350K	2%	143M	6s
4400K	2%	53.2M	6s
4450K	2%	34.5M	6s
4500K	2%	30.8M	6s
4550K	2%	28.7M	6s
4600K	2%	34.1M	6s

4650K	2%	65.3M	6s
4700K	2%	78.8M	6s
4750K	2%	28.9M	6s
4800K	2%	33.1M	6s
4850K	2%	32.4M	6s
4900K	2%	42.3M	6s
4950K	2%	49.3M	6s
5000K	2%	36.0M	6s
5050K	2%	33.4M	6s
5100K	2%	33.3M	6s
5150K	2%	51.5M	6s
5200K	2%	32.8M	6s
5250K	2%	36.2M	6s
5300K	2%	48.4M	6s
5350K	2%	39.1M	6s
5400K	2%	35.2M	6s
5450K	2%	52.2M	6s
5500K	2%	34.9M	6s
5550K	2%	52.8M	6s
5600K	3%	31.9M	6s
5650K	3%	93.6M	6s
5700K	3%	190M	6s
5750K	3%	39.9M	6s
5800K	3%	32.7M	6s
5850K	3%	35.5M	6s
5900K	3%	36.3M	6s
5950K	3%	29.8M	6s
6000K	3%	31.8M	6s
6050K	3%	48.2M	6s
6100K	3%	33.9M	6s
6150K	3%	36.8M	6s
6200K	3%	44.5M	6s
6250K	3%	49.2M	6s
6300K	3%	82.2M	6s
6350K	3%	35.2M	6s
6400K	3%	161M	5s
6450K	3%	145M	5s
6500K	3%	121M	5s
6550K	3%	139M	5s
6600K	3%	155M	5s
6650K	3%	164M	5s
6700K	3%	124M	5s
6750K	3%	122M	5s
6800K	3%	147M	5s
6850K	3%	152M	5s
6900K	3%	161M	5s
6950K	3%	135M	5s
7000K	3%	158M	5s
7050K	3%	164M	5s
7100K	3%	152M	5s
7150K	3%	120M	5s

7200K	3%	154M	5s
7250K	3%	33.4M	5s
7300K	3%	101M	5s
7350K	3%	109M	5s
7400K	3%	130M	5s
7450K	3%	33.0M	5s
7500K	4%	113M	5s
7550K	4%	92.6M	5s
7600K	4%	153M	5s
7650K	4%	86.1M	5s
7700K	4%	78.3M	5s
7750K	4%	73.6M	5s
7800K	4%	146M	5s
7850K	4%	94.6M	5s
7900K	4%	24.3M	5s
7950K	4%	89.3M	5s
8000K	4%	109M	5s
8050K	4%	80.3M	5s
8100K	4%	142M	5s
8150K	4%	98.9M	5s
8200K	4%	74.0M	5s
8250K	4%	79.1M	5s
8300K	4%	144M	5s
8350K	4%	67.1M	5s
8400K	4%	100M	5s
8450K	4%	72.0M	5s
8500K	4%	86.7M	5s
8550K	4%	134M	5s
8600K	4%	86.8M	5s
8650K	4%	75.8M	5s
8700K	4%	79.6M	4s
8750K	4%	78.0M	4s
8800K	4%	145M	4s
8850K	4%	99.3M	4s
8900K	4%	90.8M	4s
8950K	4%	81.9M	4s
9000K	4%	148M	4s
9050K	4%	93.0M	4s
9100K	4%	97.8M	4s
9150K	4%	75.3M	4s
9200K	4%	94.4M	4s
9250K	4%	146M	4s
9300K	4%	106M	4s
9350K	4%	83.1M	4s
9400K	5%	90.5M	4s
9450K	5%	147M	4s
9500K	5%	103M	4s
9550K	5%	81.4M	4s
9600K	5%	74.8M	4s
9650K	5%	94.1M	4s
9700K	5%	95.1M	4s

9750K	5%	224M	4s
9800K	5%	84.2M	4s
9850K	5%	105M	4s
9900K	5%	106M	4s
9950K	5%	88.1M	4s
10000K	5%	155M	4s
10050K	5%	116M	4s
10100K	5%	115M	4s
10150K	5%	95.7M	4s
10200K	5%	117M	4s
10250K	5%	155M	4s
10300K	5%	109M	4s
10350K	5%	95.0M	4s
10400K	5%	106M	4s
10450K	5%	162M	4s
10500K	5%	120M	4s
10550K	5%	103M	4s
10600K	5%	105M	4s
10650K	5%	108M	4s
10700K	5%	108M	4s
10750K	5%	101M	4s
10800K	5%	148M	4s
10850K	5%	111M	4s
10900K	5%	105M	4s
10950K	5%	104M	4s
11000K	5%	104M	4s
11050K	5%	113M	4s
11100K	5%	154M	4s
11150K	5%	100M	4s
11200K	5%	112M	4s
11250K	6%	111M	4s
11300K	6%	112M	4s
11350K	6%	101M	4s
11400K	6%	110M	4s
11450K	6%	113M	4s
11500K	6%	114M	4s
11550K	6%	90.2M	4s
11600K	6%	111M	4s
11650K	6%	110M	4s
11700K	6%	109M	4s
11750K	6%	106M	4s
11800K	6%	117M	4s
11850K	6%	106M	4s
11900K	6%	113M	4s
11950K	6%	99.2M	4s
12000K	6%	111M	4s
12050K	6%	114M	4s
12100K	6%	113M	4s
12150K	6%	101M	4s
12200K	6%	86.0M	4s
12250K	6%	97.9M	4s

12300K	6%	112M	4s
12350K	6%	100M	4s
12400K	6%	106M	4s
12450K	6%	80.3M	4s
12500K	6%	100M	4s
12550K	6%	120M	4s
12600K	6%	130M	4s
12650K	6%	105M	4s
12700K	6%	120M	4s
12750K	6%	87.9M	4s
12800K	6%	120M	4s
12850K	6%	124M	3s
12900K	6%	120M	3s
12950K	6%	121M	3s
13000K	6%	132M	3s
13050K	6%	114M	3s
13100K	6%	126M	3s
13150K	7%	121M	3s
13200K	7%	125M	3s
13250K	7%	129M	3s
13300K	7%	140M	3s
13350K	7%	69.6M	3s
13400K	7%	135M	3s
13450K	7%	84.5M	3s
13500K	7%	70.0M	3s
13550K	7%	101M	3s
13600K	7%	175M	3s
13650K	7%	164M	3s
13700K	7%	294M	3s
13750K	7%	156M	3s
13800K	7%	181M	3s
13850K	7%	190M	3s
13900K	7%	161M	3s
13950K	7%	192M	3s
14000K	7%	186M	3s
14050K	7%	158M	3s
14100K	7%	255M	3s
14150K	7%	93.4M	3s
14200K	7%	219M	3s
14250K	7%	167M	3s
14300K	7%	167M	3s
14350K	7%	150M	3s
14400K	7%	249M	3s
14450K	7%	134M	3s
14500K	7%	157M	3s
14550K	7%	149M	3s
14600K	7%	142M	3s
14650K	7%	165M	3s
14700K	7%	245M	3s
14750K	7%	147M	3s
14800K	7%	72.3M	3s

14850K	7%	118M	3s
14900K	7%	183M	3s
14950K	7%	101M	3s
15000K	7%	115M	3s
15050K	8%	112M	3s
15100K	8%	117M	3s
15150K	8%	79.8M	3s
15200K	8%	86.1M	3s
15250K	8%	93.7M	3s
15300K	8%	181M	3s
15350K	8%	130M	3s
15400K	8%	116M	3s
15450K	8%	95.5M	3s
15500K	8%	95.5M	3s
15550K	8%	153M	3s
15600K	8%	179M	3s
15650K	8%	99.5M	3s
15700K	8%	175M	3s
15750K	8%	66.9M	3s
15800K	8%	184M	3s
15850K	8%	187M	3s
15900K	8%	191M	3s
15950K	8%	85.0M	3s
16000K	8%	184M	3s
16050K	8%	80.0M	3s
16100K	8%	182M	3s
16150K	8%	167M	3s

*** WARNING: skipped 236740 bytes of output ***

171950K	91%	90.9M	0s
172000K	91%	80.7M	0s
172050K	91%	84.3M	0s
172100K	91%	101M	0s
172150K	91%	101M	0s
172200K	91%	159M	0s
172250K	91%	137M	0s
172300K	91%	164M	0s
172350K	91%	154M	0s
172400K	91%	163M	0s
172450K	91%	140M	0s
172500K	91%	158M	0s
172550K	91%	118M	0s
172600K	91%	91.3M	0s
172650K	91%	78.4M	0s
172700K	91%	57.9M	0s
172750K	91%	95.0M	0s
172800K	91%	137M	0s
172850K	91%	76.4M	0s
172900K	91%	115M	0s
172950K	91%	106M	0s

173000K	91%	138M	0s
173050K	91%	139M	0s
173100K	91%	166M	0s
173150K	91%	167M	0s
173200K	92%	151M	0s
173250K	92%	138M	0s
173300K	92%	158M	0s
173350K	92%	163M	0s
173400K	92%	162M	0s
173450K	92%	129M	0s
173500K	92%	158M	0s
173550K	92%	156M	0s
173600K	92%	80.0M	0s
173650K	92%	79.7M	0s
173700K	92%	96.6M	0s
173750K	92%	92.9M	0s
173800K	92%	76.5M	0s
173850K	92%	102M	0s
173900K	92%	154M	0s
173950K	92%	141M	0s
174000K	92%	166M	0s
174050K	92%	107M	0s
174100K	92%	160M	0s
174150K	92%	148M	0s
174200K	92%	128M	0s
174250K	92%	140M	0s
174300K	92%	149M	0s
174350K	92%	168M	0s
174400K	92%	163M	0s
174450K	92%	121M	0s
174500K	92%	155M	0s
174550K	92%	87.6M	0s
174600K	92%	81.2M	0s
174650K	92%	79.1M	0s
174700K	92%	85.7M	0s
174750K	92%	98.3M	0s
174800K	92%	107M	0s
174850K	92%	92.6M	0s
174900K	92%	101M	0s
174950K	92%	92.5M	0s
175000K	92%	154M	0s
175050K	93%	70.5M	0s
175100K	93%	118M	0s
175150K	93%	163M	0s
175200K	93%	162M	0s
175250K	93%	139M	0s
175300K	93%	133M	0s
175350K	93%	151M	0s
175400K	93%	127M	0s
175450K	93%	137M	0s
175500K	93%	77.9M	0s

175550K	93%	84.8M	0s
175600K	93%	114M	0s
175650K	93%	89.0M	0s
175700K	93%	95.2M	0s
175750K	93%	86.8M	0s
175800K	93%	131M	0s
175850K	93%	134M	0s
175900K	93%	117M	0s
175950K	93%	165M	0s
176000K	93%	145M	0s
176050K	93%	91.1M	0s
176100K	93%	108M	0s
176150K	93%	134M	0s
176200K	93%	153M	0s
176250K	93%	140M	0s
176300K	93%	146M	0s
176350K	93%	157M	0s
176400K	93%	164M	0s
176450K	93%	128M	0s
176500K	93%	100M	0s
176550K	93%	96.4M	0s
176600K	93%	87.2M	0s
176650K	93%	77.9M	0s
176700K	93%	87.1M	0s
176750K	93%	165M	0s
176800K	93%	101M	0s
176850K	93%	138M	0s
176900K	93%	162M	0s
176950K	94%	90.2M	0s
177000K	94%	131M	0s
177050K	94%	135M	0s
177100K	94%	81.3M	0s
177150K	94%	161M	0s
177200K	94%	144M	0s
177250K	94%	139M	0s
177300K	94%	133M	0s
177350K	94%	165M	0s
177400K	94%	155M	0s
177450K	94%	146M	0s
177500K	94%	162M	0s
177550K	94%	110M	0s
177600K	94%	94.0M	0s
177650K	94%	75.8M	0s
177700K	94%	92.6M	0s
177750K	94%	106M	0s
177800K	94%	84.0M	0s
177850K	94%	68.5M	0s
177900K	94%	161M	0s
177950K	94%	89.2M	0s
178000K	94%	115M	0s
178050K	94%	83.1M	0s

178100K	94%	96.8M	0s
178150K	94%	140M	0s
178200K	94%	156M	0s
178250K	94%	137M	0s
178300K	94%	160M	0s
178350K	94%	166M	0s
178400K	94%	163M	0s
178450K	94%	53.4M	0s
178500K	94%	58.8M	0s
178550K	94%	93.1M	0s
178600K	94%	108M	0s
178650K	94%	81.5M	0s
178700K	94%	123M	0s
178750K	94%	162M	0s
178800K	95%	164M	0s
178850K	95%	137M	0s
178900K	95%	87.5M	0s
178950K	95%	112M	0s
179000K	95%	108M	0s
179050K	95%	103M	0s
179100K	95%	119M	0s
179150K	95%	151M	0s
179200K	95%	137M	0s
179250K	95%	139M	0s
179300K	95%	156M	0s
179350K	95%	153M	0s
179400K	95%	154M	0s
179450K	95%	123M	0s
179500K	95%	78.3M	0s
179550K	95%	79.2M	0s
179600K	95%	120M	0s
179650K	95%	93.2M	0s
179700K	95%	109M	0s
179750K	95%	150M	0s
179800K	95%	159M	0s
179850K	95%	114M	0s
179900K	95%	148M	0s
179950K	95%	69.8M	0s
180000K	95%	110M	0s
180050K	95%	77.4M	0s
180100K	95%	153M	0s
180150K	95%	154M	0s
180200K	95%	121M	0s
180250K	95%	139M	0s
180300K	95%	124M	0s
180350K	95%	151M	0s
180400K	95%	119M	0s
180450K	95%	84.7M	0s
180500K	95%	93.1M	0s
180550K	95%	80.8M	0s
180600K	95%	103M	0s

180650K	95%	67.2M	0s
180700K	96%	81.5M	0s
180750K	96%	155M	0s
180800K	96%	164M	0s
180850K	96%	83.9M	0s
180900K	96%	76.2M	0s
180950K	96%	113M	0s
181000K	96%	78.5M	0s
181050K	96%	131M	0s
181100K	96%	146M	0s
181150K	96%	127M	0s
181200K	96%	162M	0s
181250K	96%	140M	0s
181300K	96%	147M	0s
181350K	96%	133M	0s
181400K	96%	87.2M	0s
181450K	96%	65.6M	0s
181500K	96%	103M	0s
181550K	96%	90.3M	0s
181600K	96%	117M	0s
181650K	96%	109M	0s
181700K	96%	123M	0s
181750K	96%	150M	0s
181800K	96%	77.0M	0s
181850K	96%	115M	0s
181900K	96%	140M	0s
181950K	96%	150M	0s
182000K	96%	164M	0s
182050K	96%	109M	0s
182100K	96%	165M	0s
182150K	96%	159M	0s
182200K	96%	159M	0s
182250K	96%	97.3M	0s
182300K	96%	164M	0s
182350K	96%	139M	0s
182400K	96%	112M	0s
182450K	96%	79.6M	0s
182500K	96%	88.5M	0s
182550K	96%	118M	0s
182600K	97%	88.0M	0s
182650K	97%	110M	0s
182700K	97%	172M	0s
182750K	97%	125M	0s
182800K	97%	150M	0s
182850K	97%	110M	0s
182900K	97%	60.5M	0s
182950K	97%	97.4M	0s
183000K	97%	148M	0s
183050K	97%	135M	0s
183100K	97%	164M	0s
183150K	97%	153M	0s

183200K	97%	138M	0s
183250K	97%	120M	0s
183300K	97%	132M	0s
183350K	97%	153M	0s
183400K	97%	151M	0s
183450K	97%	66.3M	0s
183500K	97%	114M	0s
183550K	97%	77.2M	0s
183600K	97%	73.1M	0s
183650K	97%	19.3M	0s
183700K	97%	32.5M	0s
183750K	97%	44.6M	0s
183800K	97%	31.6M	0s
183850K	97%	9.97M	0s
183900K	97%	61.4M	0s
183950K	97%	162M	0s
184000K	97%	165M	0s
184050K	97%	110M	0s
184100K	97%	166M	0s
184150K	97%	156M	0s
184200K	97%	152M	0s
184250K	97%	138M	0s
184300K	97%	166M	0s
184350K	97%	149M	0s
184400K	97%	111M	0s
184450K	98%	107M	0s
184500K	98%	155M	0s
184550K	98%	155M	0s
184600K	98%	152M	0s
184650K	98%	142M	0s
184700K	98%	132M	0s
184750K	98%	122M	0s
184800K	98%	165M	0s
184850K	98%	127M	0s
184900K	98%	156M	0s
184950K	98%	156M	0s
185000K	98%	61.7M	0s
185050K	98%	80.4M	0s
185100K	98%	145M	0s
185150K	98%	154M	0s
185200K	98%	166M	0s
185250K	98%	130M	0s
185300K	98%	155M	0s
185350K	98%	155M	0s
185400K	98%	149M	0s
185450K	98%	140M	0s
185500K	98%	154M	0s
185550K	98%	168M	0s
185600K	98%	73.5M	0s
185650K	98%	141M	0s
185700K	98%	157M	0s

185750K	98%	156M	0s
185800K	98%	153M	0s
185850K	98%	140M	0s
185900K	98%	160M	0s
185950K	98%	166M	0s
186000K	98%	166M	0s
186050K	98%	138M	0s
186100K	98%	139M	0s
186150K	98%	94.5M	0s
186200K	98%	82.3M	0s
186250K	98%	79.1M	0s
186300K	98%	122M	0s
186350K	99%	77.5M	0s
186400K	99%	88.4M	0s
186450K	99%	142M	0s
186500K	99%	153M	0s
186550K	99%	155M	0s
186600K	99%	146M	0s
186650K	99%	39.7M	0s
186700K	99%	118M	0s
186750K	99%	93.6M	0s
186800K	99%	163M	0s
186850K	99%	122M	0s
186900K	99%	152M	0s
186950K	99%	164M	0s
187000K	99%	153M	0s
187050K	99%	139M	0s
187100K	99%	87.8M	0s
187150K	99%	127M	0s
187200K	99%	85.3M	0s
187250K	99%	111M	0s
187300K	99%	75.7M	0s
187350K	99%	125M	0s
187400K	99%	121M	0s
187450K	99%	114M	0s
187500K	99%	128M	0s
187550K	99%	163M	0s
187600K	99%	157M	0s
187650K	99%	139M	0s
187700K	99%	159M	0s
187750K	99%	155M	0s
187800K	99%	128M	0s
187850K	99%	144M	0s
187900K	99%	168M	0s
187950K	99%	153M	0s
188000K	99%	161M	0s
188050K	99%	141M	0s
188100K	99%	160M	0s
188150K	99%	154M	0s
188200K	99%	136M	0s
188250K				100%	121M=1.8s	

2020-05-31 14:07:48 (103 MB/s) - 'title.akas.tsv.gz' saved [192780293/192780293]

/bin/bash: line 1: fg: no job control

```
In [6]: title_akas = spark.read.option("sep", "\t").csv('file:/databricks/driver/titl
e.akas.tsv', header=True, inferSchema = True)
title_akas.cache()
title_akas.show(3)
```

```
+-----+-----+-----+-----+-----+-----+
-----+
titleId|ordering|          title|region|language|      types|  attributes|i
sOriginalTitle|
+-----+-----+-----+-----+-----+-----+
-----+
tt0000001|      1|      Карменсита|    UA|      \N|imdbDisplay|      \N|
0|
tt0000001|      2|      Carmencita|    DE|      \N|      \N|literal title|
0|
tt0000001|      3|Carmencita - span...|    HU|      \N|imdbDisplay|      \N|
0|
+-----+-----+-----+-----+-----+-----+
-----+
only showing top 3 rows
```

```
In [7]: %sh wget https://datasets.imdbws.com/title.basics.tsv.gz  
%sh  
gunzip title.basics.tsv.gz
```

```
--2020-05-31 14:08:56-- https://datasets.imdbws.com/title.basics.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.37, 13.224.13.54,
13.224.13.26, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.37|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 121814992 (116M) [binary/octet-stream]
Saving to: 'title.basics.tsv.gz'
```

```
  0K ..... 0% 5.02M 23s
 50K ..... 0% 9.36M 18s
100K ..... 0% 13.1M 15s
150K ..... 0% 19.1M 13s
200K ..... 0% 16.4M 12s
250K ..... 0% 31.5M 10s
300K ..... 0% 36.4M 9s
350K ..... 0% 21.3M 9s
400K ..... 0% 46.9M 8s
450K ..... 0% 35.4M 8s
500K ..... 0% 55.5M 7s
550K ..... 0% 51.1M 7s
600K ..... 0% 54.6M 6s
650K ..... 0% 59.6M 6s
700K ..... 0% 33.5M 6s
750K ..... 0% 47.8M 6s
800K ..... 0% 138M 5s
850K ..... 0% 136M 5s
900K ..... 0% 183M 5s
950K ..... 0% 116M 5s
1000K ..... 0% 101M 4s
1050K ..... 0% 77.9M 4s
1100K ..... 0% 89.2M 4s
1150K ..... 1% 77.7M 4s
1200K ..... 1% 103M 4s
1250K ..... 1% 138M 4s
1300K ..... 1% 90.9M 4s
1350K ..... 1% 103M 4s
1400K ..... 1% 78.6M 4s
1450K ..... 1% 156M 3s
1500K ..... 1% 141M 3s
1550K ..... 1% 135M 3s
1600K ..... 1% 126M 3s
1650K ..... 1% 149M 3s
1700K ..... 1% 281M 3s
1750K ..... 1% 177M 3s
1800K ..... 1% 106M 3s
1850K ..... 1% 185M 3s
1900K ..... 1% 125M 3s
1950K ..... 1% 129M 3s
2000K ..... 1% 156M 3s
2050K ..... 1% 174M 3s
```

2100K	1%	270M	3s
2150K	1%	206M	3s
2200K	1%	187M	3s
2250K	1%	243M	2s
2300K	1%	200M	2s
2350K	2%	179M	2s
2400K	2%	214M	2s
2450K	2%	192M	2s
2500K	2%	200M	2s
2550K	2%	170M	2s
2600K	2%	195M	2s
2650K	2%	168M	2s
2700K	2%	172M	2s
2750K	2%	162M	2s
2800K	2%	202M	2s
2850K	2%	206M	2s
2900K	2%	199M	2s
2950K	2%	237M	2s
3000K	2%	309M	2s
3050K	2%	191M	2s
3100K	2%	302M	2s
3150K	2%	179M	2s
3200K	2%	232M	2s
3250K	2%	241M	2s
3300K	2%	196M	2s
3350K	2%	276M	2s
3400K	2%	81.4M	2s
3450K	2%	133M	2s
3500K	2%	147M	2s
3550K	3%	120M	2s
3600K	3%	153M	2s
3650K	3%	153M	2s
3700K	3%	117M	2s
3750K	3%	78.6M	2s
3800K	3%	154M	2s
3850K	3%	179M	2s
3900K	3%	94.7M	2s
3950K	3%	127M	2s
4000K	3%	108M	2s
4050K	3%	137M	2s
4100K	3%	148M	2s
4150K	3%	127M	2s
4200K	3%	164M	2s
4250K	3%	140M	2s
4300K	3%	136M	2s
4350K	3%	129M	2s
4400K	3%	146M	2s
4450K	3%	147M	2s
4500K	3%	142M	2s
4550K	3%	137M	2s
4600K	3%	149M	2s

4650K	3%	145M	2s
4700K	3%	127M	2s
4750K	4%	89.2M	2s
4800K	4%	191M	2s
4850K	4%	131M	2s
4900K	4%	160M	2s
4950K	4%	141M	2s
5000K	4%	162M	2s
5050K	4%	146M	1s
5100K	4%	171M	1s
5150K	4%	145M	1s
5200K	4%	147M	1s
5250K	4%	140M	1s
5300K	4%	182M	1s
5350K	4%	150M	1s
5400K	4%	149M	1s
5450K	4%	160M	1s
5500K	4%	147M	1s
5550K	4%	147M	1s
5600K	4%	148M	1s
5650K	4%	178M	1s
5700K	4%	96.2M	1s
5750K	4%	124M	1s
5800K	4%	159M	1s
5850K	4%	129M	1s
5900K	5%	140M	1s
5950K	5%	81.6M	1s
6000K	5%	141M	1s
6050K	5%	143M	1s
6100K	5%	99.8M	1s
6150K	5%	157M	1s
6200K	5%	95.3M	1s
6250K	5%	128M	1s
6300K	5%	136M	1s
6350K	5%	73.8M	1s
6400K	5%	149M	1s
6450K	5%	119M	1s
6500K	5%	141M	1s
6550K	5%	115M	1s
6600K	5%	125M	1s
6650K	5%	154M	1s
6700K	5%	126M	1s
6750K	5%	105M	1s
6800K	5%	149M	1s
6850K	5%	155M	1s
6900K	5%	147M	1s
6950K	5%	127M	1s
7000K	5%	154M	1s
7050K	5%	148M	1s
7100K	6%	142M	1s
7150K	6%	121M	1s

7200K	6%	149M	1s
7250K	6%	152M	1s
7300K	6%	147M	1s
7350K	6%	136M	1s
7400K	6%	146M	1s
7450K	6%	156M	1s
7500K	6%	140M	1s
7550K	6%	127M	1s
7600K	6%	154M	1s
7650K	6%	139M	1s
7700K	6%	132M	1s
7750K	6%	142M	1s
7800K	6%	133M	1s
7850K	6%	125M	1s
7900K	6%	146M	1s
7950K	6%	109M	1s
8000K	6%	99.8M	1s
8050K	6%	91.8M	1s
8100K	6%	142M	1s
8150K	6%	122M	1s
8200K	6%	144M	1s
8250K	6%	137M	1s
8300K	7%	158M	1s
8350K	7%	93.2M	1s
8400K	7%	148M	1s
8450K	7%	134M	1s
8500K	7%	122M	1s
8550K	7%	124M	1s
8600K	7%	140M	1s
8650K	7%	124M	1s
8700K	7%	126M	1s
8750K	7%	124M	1s
8800K	7%	174M	1s
8850K	7%	85.9M	1s
8900K	7%	154M	1s
8950K	7%	103M	1s
9000K	7%	130M	1s
9050K	7%	157M	1s
9100K	7%	99.6M	1s
9150K	7%	127M	1s
9200K	7%	158M	1s
9250K	7%	154M	1s
9300K	7%	157M	1s
9350K	7%	54.7M	1s
9400K	7%	49.7M	1s
9450K	7%	86.1M	1s
9500K	8%	132M	1s
9550K	8%	112M	1s
9600K	8%	89.7M	1s
9650K	8%	110M	1s
9700K	8%	138M	1s

9750K	8%	149M	1s
9800K	8%	73.9M	1s
9850K	8%	192M	1s
9900K	8%	84.9M	1s
9950K	8%	97.6M	1s
10000K	8%	163M	1s
10050K	8%	143M	1s
10100K	8%	73.9M	1s
10150K	8%	95.8M	1s
10200K	8%	198M	1s
10250K	8%	190M	1s
10300K	8%	193M	1s
10350K	8%	209M	1s
10400K	8%	285M	1s
10450K	8%	272M	1s
10500K	8%	305M	1s
10550K	8%	153M	1s
10600K	8%	269M	1s
10650K	8%	275M	1s
10700K	9%	232M	1s
10750K	9%	129M	1s
10800K	9%	299M	1s
10850K	9%	272M	1s
10900K	9%	203M	1s
10950K	9%	171M	1s
11000K	9%	7.16M	1s
11050K	9%	42.5M	1s
11100K	9%	26.4M	1s
11150K	9%	108M	1s
11200K	9%	165M	1s
11250K	9%	154M	1s
11300K	9%	161M	1s
11350K	9%	139M	1s
11400K	9%	158M	1s
11450K	9%	141M	1s
11500K	9%	161M	1s
11550K	9%	131M	1s
11600K	9%	161M	1s
11650K	9%	140M	1s
11700K	9%	161M	1s
11750K	9%	104M	1s
11800K	9%	162M	1s
11850K	10%	112M	1s
11900K	10%	112M	1s
11950K	10%	130M	1s
12000K	10%	133M	1s
12050K	10%	47.4M	1s
12100K	10%	82.8M	1s
12150K	10%	68.1M	1s
12200K	10%	88.6M	1s
12250K	10%	90.2M	1s

12300K	10%	103M	1s
12350K	10%	135M	1s
12400K	10%	164M	1s
12450K	10%	144M	1s
12500K	10%	159M	1s
12550K	10%	144M	1s
12600K	10%	91.3M	1s
12650K	10%	149M	1s
12700K	10%	167M	1s
12750K	10%	140M	1s
12800K	10%	147M	1s
12850K	10%	162M	1s
12900K	10%	157M	1s
12950K	10%	136M	1s
13000K	10%	159M	1s
13050K	11%	64.9M	1s
13100K	11%	68.8M	1s
13150K	11%	75.9M	1s
13200K	11%	70.6M	1s
13250K	11%	68.5M	1s
13300K	11%	93.0M	1s
13350K	11%	139M	1s
13400K	11%	93.8M	1s
13450K	11%	67.7M	1s
13500K	11%	44.3M	1s
13550K	11%	32.1M	1s
13600K	11%	87.8M	1s
13650K	11%	159M	1s
13700K	11%	142M	1s
13750K	11%	135M	1s
13800K	11%	156M	1s
13850K	11%	154M	1s
13900K	11%	123M	1s
13950K	11%	127M	1s
14000K	11%	161M	1s
14050K	11%	139M	1s
14100K	11%	150M	1s
14150K	11%	124M	1s
14200K	11%	157M	1s
14250K	12%	139M	1s
14300K	12%	145M	1s
14350K	12%	110M	1s
14400K	12%	160M	1s
14450K	12%	158M	1s
14500K	12%	147M	1s
14550K	12%	133M	1s
14600K	12%	109M	1s
14650K	12%	144M	1s
14700K	12%	126M	1s
14750K	12%	138M	1s
14800K	12%	75.5M	1s

14850K	12%	160M	1s
14900K	12%	154M	1s
14950K	12%	79.3M	1s
15000K	12%	167M	1s
15050K	12%	104M	1s
15100K	12%	155M	1s
15150K	12%	137M	1s
15200K	12%	149M	1s
15250K	12%	77.4M	1s
15300K	12%	73.1M	1s
15350K	12%	78.0M	1s
15400K	12%	194M	1s
15450K	13%	276M	1s
15500K	13%	296M	1s
15550K	13%	89.7M	1s
15600K	13%	68.8M	1s
15650K	13%	87.2M	1s
15700K	13%	66.7M	1s
15750K	13%	81.7M	1s
15800K	13%	90.8M	1s
15850K	13%	104M	1s
15900K	13%	155M	1s
15950K	13%	201M	1s
16000K	13%	302M	1s
16050K	13%	300M	1s
16100K	13%	274M	1s
16150K	13%	121M	1s

*** WARNING: skipped 131404 bytes of output ***

102650K	86%	158M	0s
102700K	86%	166M	0s
102750K	86%	194M	0s
102800K	86%	167M	0s
102850K	86%	173M	0s
102900K	86%	177M	0s
102950K	86%	175M	0s
103000K	86%	181M	0s
103050K	86%	134M	0s
103100K	86%	163M	0s
103150K	86%	185M	0s
103200K	86%	180M	0s
103250K	86%	169M	0s
103300K	86%	190M	0s
103350K	86%	182M	0s
103400K	86%	181M	0s
103450K	87%	159M	0s
103500K	87%	171M	0s
103550K	87%	195M	0s
103600K	87%	191M	0s
103650K	87%	155M	0s

103700K	87%	167M	0s
103750K	87%	157M	0s
103800K	87%	194M	0s
103850K	87%	137M	0s
103900K	87%	168M	0s
103950K	87%	174M	0s
104000K	87%	184M	0s
104050K	87%	148M	0s
104100K	87%	184M	0s
104150K	87%	112M	0s
104200K	87%	177M	0s
104250K	87%	140M	0s
104300K	87%	167M	0s
104350K	87%	163M	0s
104400K	87%	175M	0s
104450K	87%	161M	0s
104500K	87%	183M	0s
104550K	87%	188M	0s
104600K	87%	180M	0s
104650K	88%	155M	0s
104700K	88%	187M	0s
104750K	88%	179M	0s
104800K	88%	192M	0s
104850K	88%	164M	0s
104900K	88%	169M	0s
104950K	88%	171M	0s
105000K	88%	134M	0s
105050K	88%	150M	0s
105100K	88%	154M	0s
105150K	88%	166M	0s
105200K	88%	161M	0s
105250K	88%	161M	0s
105300K	88%	157M	0s
105350K	88%	161M	0s
105400K	88%	184M	0s
105450K	88%	146M	0s
105500K	88%	165M	0s
105550K	88%	133M	0s
105600K	88%	184M	0s
105650K	88%	158M	0s
105700K	88%	170M	0s
105750K	88%	179M	0s
105800K	88%	182M	0s
105850K	89%	157M	0s
105900K	89%	181M	0s
105950K	89%	191M	0s
106000K	89%	187M	0s
106050K	89%	174M	0s
106100K	89%	187M	0s
106150K	89%	176M	0s
106200K	89%	169M	0s

106250K	89%	138M	0s
106300K	89%	161M	0s
106350K	89%	178M	0s
106400K	89%	173M	0s
106450K	89%	163M	0s
106500K	89%	176M	0s
106550K	89%	192M	0s
106600K	89%	187M	0s
106650K	89%	141M	0s
106700K	89%	179M	0s
106750K	89%	167M	0s
106800K	89%	171M	0s
106850K	89%	155M	0s
106900K	89%	171M	0s
106950K	89%	178M	0s
107000K	89%	167M	0s
107050K	90%	152M	0s
107100K	90%	183M	0s
107150K	90%	191M	0s
107200K	90%	188M	0s
107250K	90%	171M	0s
107300K	90%	187M	0s
107350K	90%	192M	0s
107400K	90%	173M	0s
107450K	90%	152M	0s
107500K	90%	186M	0s
107550K	90%	188M	0s
107600K	90%	188M	0s
107650K	90%	152M	0s
107700K	90%	170M	0s
107750K	90%	190M	0s
107800K	90%	176M	0s
107850K	90%	137M	0s
107900K	90%	176M	0s
107950K	90%	189M	0s
108000K	90%	193M	0s
108050K	90%	162M	0s
108100K	90%	181M	0s
108150K	90%	174M	0s
108200K	90%	189M	0s
108250K	91%	144M	0s
108300K	91%	151M	0s
108350K	91%	182M	0s
108400K	91%	180M	0s
108450K	91%	166M	0s
108500K	91%	191M	0s
108550K	91%	194M	0s
108600K	91%	180M	0s
108650K	91%	150M	0s
108700K	91%	188M	0s
108750K	91%	175M	0s

108800K	91%	196M	0s
108850K	91%	174M	0s
108900K	91%	174M	0s
108950K	91%	192M	0s
109000K	91%	185M	0s
109050K	91%	147M	0s
109100K	91%	183M	0s
109150K	91%	192M	0s
109200K	91%	191M	0s
109250K	91%	168M	0s
109300K	91%	182M	0s
109350K	91%	170M	0s
109400K	92%	168M	0s
109450K	92%	134M	0s
109500K	92%	177M	0s
109550K	92%	186M	0s
109600K	92%	161M	0s
109650K	92%	166M	0s
109700K	92%	166M	0s
109750K	92%	172M	0s
109800K	92%	190M	0s
109850K	92%	151M	0s
109900K	92%	193M	0s
109950K	92%	186M	0s
110000K	92%	199M	0s
110050K	92%	167M	0s
110100K	92%	186M	0s
110150K	92%	187M	0s
110200K	92%	185M	0s
110250K	92%	153M	0s
110300K	92%	182M	0s
110350K	92%	179M	0s
110400K	92%	181M	0s
110450K	92%	160M	0s
110500K	92%	181M	0s
110550K	92%	191M	0s
110600K	93%	172M	0s
110650K	93%	144M	0s
110700K	93%	175M	0s
110750K	93%	168M	0s
110800K	93%	175M	0s
110850K	93%	135M	0s
110900K	93%	190M	0s
110950K	93%	170M	0s
111000K	93%	167M	0s
111050K	93%	142M	0s
111100K	93%	187M	0s
111150K	93%	168M	0s
111200K	93%	194M	0s
111250K	93%	172M	0s
111300K	93%	183M	0s

111350K	93%	185M	0s
111400K	93%	183M	0s
111450K	93%	148M	0s
111500K	93%	181M	0s
111550K	93%	192M	0s
111600K	93%	182M	0s
111650K	93%	181M	0s
111700K	93%	183M	0s
111750K	93%	186M	0s
111800K	94%	185M	0s
111850K	94%	156M	0s
111900K	94%	178M	0s
111950K	94%	176M	0s
112000K	94%	176M	0s
112050K	94%	170M	0s
112100K	94%	175M	0s
112150K	94%	168M	0s
112200K	94%	173M	0s
112250K	94%	143M	0s
112300K	94%	159M	0s
112350K	94%	168M	0s
112400K	94%	165M	0s
112450K	94%	162M	0s
112500K	94%	189M	0s
112550K	94%	187M	0s
112600K	94%	173M	0s
112650K	94%	157M	0s
112700K	94%	190M	0s
112750K	94%	190M	0s
112800K	94%	193M	0s
112850K	94%	163M	0s
112900K	94%	171M	0s
112950K	94%	190M	0s
113000K	95%	178M	0s
113050K	95%	159M	0s
113100K	95%	183M	0s
113150K	95%	178M	0s
113200K	95%	183M	0s
113250K	95%	171M	0s
113300K	95%	173M	0s
113350K	95%	190M	0s
113400K	95%	189M	0s
113450K	95%	157M	0s
113500K	95%	180M	0s
113550K	95%	182M	0s
113600K	95%	186M	0s
113650K	95%	142M	0s
113700K	95%	169M	0s
113750K	95%	175M	0s
113800K	95%	165M	0s
113850K	95%	144M	0s

113900K	95%	183M	0s
113950K	95%	185M	0s
114000K	95%	177M	0s
114050K	95%	162M	0s
114100K	95%	174M	0s
114150K	95%	189M	0s
114200K	96%	181M	0s
114250K	96%	154M	0s
114300K	96%	179M	0s
114350K	96%	188M	0s
114400K	96%	174M	0s
114450K	96%	177M	0s
114500K	96%	173M	0s
114550K	96%	178M	0s
114600K	96%	177M	0s
114650K	96%	148M	0s
114700K	96%	168M	0s
114750K	96%	186M	0s
114800K	96%	178M	0s
114850K	96%	149M	0s
114900K	96%	174M	0s
114950K	96%	176M	0s
115000K	96%	160M	0s
115050K	96%	138M	0s
115100K	96%	155M	0s
115150K	96%	173M	0s
115200K	96%	185M	0s
115250K	96%	167M	0s
115300K	96%	178M	0s
115350K	97%	195M	0s
115400K	97%	183M	0s
115450K	97%	149M	0s
115500K	97%	186M	0s
115550K	97%	187M	0s
115600K	97%	189M	0s
115650K	97%	180M	0s
115700K	97%	178M	0s
115750K	97%	184M	0s
115800K	97%	193M	0s
115850K	97%	147M	0s
115900K	97%	189M	0s
115950K	97%	183M	0s
116000K	97%	178M	0s
116050K	97%	166M	0s
116100K	97%	157M	0s
116150K	97%	168M	0s
116200K	97%	154M	0s
116250K	97%	143M	0s
116300K	97%	174M	0s
116350K	97%	163M	0s
116400K	97%	167M	0s

116450K	97%	156M	0s
116500K	97%	207M	0s
116550K	98%	153M	0s
116600K	98%	188M	0s
116650K	98%	183M	0s
116700K	98%	191M	0s
116750K	98%	176M	0s
116800K	98%	181M	0s
116850K	98%	189M	0s
116900K	98%	181M	0s
116950K	98%	150M	0s
117000K	98%	194M	0s
117050K	98%	183M	0s
117100K	98%	190M	0s
117150K	98%	145M	0s
117200K	98%	166M	0s
117250K	98%	175M	0s
117300K	98%	194M	0s
117350K	98%	153M	0s
117400K	98%	184M	0s
117450K	98%	180M	0s
117500K	98%	169M	0s
117550K	98%	136M	0s
117600K	98%	150M	0s
117650K	98%	160M	0s
117700K	98%	160M	0s
117750K	99%	137M	0s
117800K	99%	179M	0s
117850K	99%	185M	0s
117900K	99%	183M	0s
117950K	99%	169M	0s
118000K	99%	193M	0s
118050K	99%	178M	0s
118100K	99%	188M	0s
118150K	99%	162M	0s
118200K	99%	170M	0s
118250K	99%	188M	0s
118300K	99%	189M	0s
118350K	99%	164M	0s
118400K	99%	188M	0s
118450K	99%	186M	0s
118500K	99%	177M	0s
118550K	99%	154M	0s
118600K	99%	192M	0s
118650K	99%	187M	0s
118700K	99%	187M	0s
118750K	99%	164M	0s
118800K	99%	184M	0s
118850K	99%	182M	0s
118900K	99%	171M	0s
118950K	100%	89.2M	=0.8s

2020-05-31 14:08:57 (139 MB/s) - 'title.basics.tsv.gz' saved [121814992/121814992]

/bin/bash: line 1: fg: no job control

```
In [8]: title_basics = spark.read.option("sep", "\t").csv('file:/databricks/driver/title.basics.tsv', header=True, inferSchema = True)
title_basics.cache()
title_basics.show(3)
```

```
+-----+-----+-----+-----+-----+-----+
+-----+
      tconst|titleType|      primaryTitle|      originalTitle|isAdult|startYear|end
Year|runtimeMinutes|      genres|
+-----+-----+-----+-----+-----+-----+
+-----+
tt0000001|   short|      Carmencita|      Carmencita|    0|    1894|
\N|         1| Documentary,Short|
tt0000002|   short|Le clown et ses c...|Le clown et ses c...|    0|    1892|
\N|         5| Animation,Short|
tt0000003|   short|      Pauvre Pierrot|      Pauvre Pierrot|    0|    1892|
\N|         4| Animation,Comedy,...|
+-----+-----+-----+-----+-----+-----+
+-----+
only showing top 3 rows
```



```
In [9]: %sh wget https://datasets.imdbws.com/title.crew.tsv.gz  
%sh  
gunzip title.crew.tsv.gz
```

```
--2020-05-31 14:09:25-- https://datasets.imdbws.com/title.crew.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.32, 13.224.13.37,
13.224.13.54, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.32|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 48431137 (46M) [binary/octet-stream]
Saving to: 'title.crew.tsv.gz'
```

0K	0%	4.52M	10s
50K	0%	9.49M	8s
100K	0%	15.5M	6s
150K	0%	13.5M	5s
200K	0%	22.2M	5s
250K	0%	26.9M	4s
300K	0%	27.3M	4s
350K	0%	26.0M	4s
400K	0%	33.5M	3s
450K	1%	35.9M	3s
500K	1%	35.6M	3s
550K	1%	30.8M	3s
600K	1%	35.0M	3s
650K	1%	30.3M	3s
700K	1%	34.2M	3s
750K	1%	31.0M	2s
800K	1%	30.3M	2s
850K	1%	32.9M	2s
900K	2%	36.8M	2s
950K	2%	29.5M	2s
1000K	2%	37.8M	2s
1050K	2%	36.2M	2s
1100K	2%	36.1M	2s
1150K	2%	30.2M	2s
1200K	2%	36.4M	2s
1250K	2%	34.7M	2s
1300K	2%	35.6M	2s
1350K	2%	31.0M	2s
1400K	3%	36.0M	2s
1450K	3%	37.0M	2s
1500K	3%	22.2M	2s
1550K	3%	30.3M	2s
1600K	3%	36.4M	2s
1650K	3%	36.9M	2s
1700K	3%	35.1M	2s
1750K	3%	31.6M	2s
1800K	3%	36.5M	2s
1850K	4%	36.2M	2s
1900K	4%	32.7M	2s
1950K	4%	31.3M	2s
2000K	4%	35.3M	2s
2050K	4%	32.9M	2s

2100K	4%	34.3M	2s
2150K	4%	58.0M	2s
2200K	4%	33.2M	2s
2250K	4%	87.6M	2s
2300K	4%	33.2M	2s
2350K	5%	31.6M	2s
2400K	5%	48.7M	2s
2450K	5%	42.5M	2s
2500K	5%	46.5M	2s
2550K	5%	174M	2s
2600K	5%	92.5M	2s
2650K	5%	35.7M	2s
2700K	5%	36.0M	2s
2750K	5%	32.9M	2s
2800K	6%	37.1M	2s
2850K	6%	34.4M	2s
2900K	6%	32.7M	2s
2950K	6%	30.9M	1s
3000K	6%	60.0M	1s
3050K	6%	41.3M	1s
3100K	6%	29.2M	1s
3150K	6%	32.6M	1s
3200K	6%	184M	1s
3250K	6%	26.7M	1s
3300K	7%	38.4M	1s
3350K	7%	184M	1s
3400K	7%	27.9M	1s
3450K	7%	33.1M	1s
3500K	7%	38.4M	1s
3550K	7%	68.6M	1s
3600K	7%	52.1M	1s
3650K	7%	34.1M	1s
3700K	7%	35.5M	1s
3750K	8%	34.3M	1s
3800K	8%	60.1M	1s
3850K	8%	35.3M	1s
3900K	8%	29.6M	1s
3950K	8%	35.0M	1s
4000K	8%	47.0M	1s
4050K	8%	60.8M	1s
4100K	8%	187M	1s
4150K	8%	59.0M	1s
4200K	8%	37.7M	1s
4250K	9%	29.2M	1s
4300K	9%	37.3M	1s
4350K	9%	37.1M	1s
4400K	9%	37.0M	1s
4450K	9%	29.5M	1s
4500K	9%	36.5M	1s
4550K	9%	30.8M	1s
4600K	9%	36.4M	1s

4650K	9%	34.3M	1s
4700K	10%	66.4M	1s
4750K	10%	54.2M	1s
4800K	10%	36.7M	1s
4850K	10%	31.3M	1s
4900K	10%	44.2M	1s
4950K	10%	34.8M	1s
5000K	10%	145M	1s
5050K	10%	54.0M	1s
5100K	10%	36.7M	1s
5150K	10%	36.6M	1s
5200K	11%	37.4M	1s
5250K	11%	33.4M	1s
5300K	11%	34.5M	1s
5350K	11%	46.7M	1s
5400K	11%	34.9M	1s
5450K	11%	63.0M	1s
5500K	11%	170M	1s
5550K	11%	59.0M	1s
5600K	11%	35.7M	1s
5650K	12%	30.0M	1s
5700K	12%	32.8M	1s
5750K	12%	35.8M	1s
5800K	12%	38.3M	1s
5850K	12%	31.3M	1s
5900K	12%	5.79M	1s
5950K	12%	36.3M	1s
6000K	12%	75.9M	1s
6050K	12%	158M	1s
6100K	13%	33.6M	1s
6150K	13%	35.4M	1s
6200K	13%	36.1M	1s
6250K	13%	29.1M	1s
6300K	13%	35.6M	1s
6350K	13%	33.3M	1s
6400K	13%	35.8M	1s
6450K	13%	42.4M	1s
6500K	13%	43.9M	1s
6550K	13%	45.3M	1s
6600K	14%	45.3M	1s
6650K	14%	28.7M	1s
6700K	14%	36.2M	1s
6750K	14%	89.3M	1s
6800K	14%	16.4M	1s
6850K	14%	31.6M	1s
6900K	14%	42.2M	1s
6950K	14%	44.0M	1s
7000K	14%	43.4M	1s
7050K	15%	33.4M	1s
7100K	15%	51.0M	1s
7150K	15%	52.3M	1s

7200K	15%	41.7M	1s
7250K	15%	32.2M	1s
7300K	15%	35.8M	1s
7350K	15%	43.4M	1s
7400K	15%	193M	1s
7450K	15%	74.9M	1s
7500K	15%	36.5M	1s
7550K	16%	36.5M	1s
7600K	16%	35.1M	1s
7650K	16%	30.0M	1s
7700K	16%	37.5M	1s
7750K	16%	41.1M	1s
7800K	16%	182M	1s
7850K	16%	33.0M	1s
7900K	16%	34.1M	1s
7950K	16%	36.8M	1s
8000K	17%	24.9M	1s
8050K	17%	34.5M	1s
8100K	17%	34.9M	1s
8150K	17%	35.9M	1s
8200K	17%	89.6M	1s
8250K	17%	164M	1s
8300K	17%	51.6M	1s
8350K	17%	38.4M	1s
8400K	17%	36.9M	1s
8450K	17%	29.9M	1s
8500K	18%	35.9M	1s
8550K	18%	38.7M	1s
8600K	18%	34.9M	1s
8650K	18%	30.2M	1s
8700K	18%	41.1M	1s
8750K	18%	37.2M	1s
8800K	18%	41.9M	1s
8850K	18%	42.2M	1s
8900K	18%	46.7M	1s
8950K	19%	32.7M	1s
9000K	19%	49.7M	1s
9050K	19%	35.0M	1s
9100K	19%	38.4M	1s
9150K	19%	84.0M	1s
9200K	19%	50.5M	1s
9250K	19%	28.5M	1s
9300K	19%	39.1M	1s
9350K	19%	86.2M	1s
9400K	19%	50.6M	1s
9450K	20%	26.7M	1s
9500K	20%	31.8M	1s
9550K	20%	29.3M	1s
9600K	20%	80.0M	1s
9650K	20%	65.1M	1s
9700K	20%	31.1M	1s

9750K	20%	30.1M	1s
9800K	20%	30.6M	1s
9850K	20%	27.3M	1s
9900K	21%	64.7M	1s
9950K	21%	42.9M	1s
10000K	21%	29.7M	1s
10050K	21%	30.3M	1s
10100K	21%	32.0M	1s
10150K	21%	92.2M	1s
10200K	21%	186M	1s
10250K	21%	41.7M	1s
10300K	21%	30.7M	1s
10350K	21%	34.2M	1s
10400K	22%	34.1M	1s
10450K	22%	28.9M	1s
10500K	22%	45.0M	1s
10550K	22%	41.0M	1s
10600K	22%	43.3M	1s
10650K	22%	30.4M	1s
10700K	22%	57.8M	1s
10750K	22%	86.2M	1s
10800K	22%	84.8M	1s
10850K	23%	178M	1s
10900K	23%	53.2M	1s
10950K	23%	32.3M	1s
11000K	23%	32.9M	1s
11050K	23%	30.8M	1s
11100K	23%	30.3M	1s
11150K	23%	33.1M	1s
11200K	23%	43.3M	1s
11250K	23%	29.7M	1s
11300K	23%	32.6M	1s
11350K	24%	132M	1s
11400K	24%	196M	1s
11450K	24%	37.3M	1s
11500K	24%	32.3M	1s
11550K	24%	36.9M	1s
11600K	24%	35.3M	1s
11650K	24%	28.0M	1s
11700K	24%	109M	1s
11750K	24%	66.1M	1s
11800K	25%	34.3M	1s
11850K	25%	28.5M	1s
11900K	25%	32.6M	1s
11950K	25%	34.3M	1s
12000K	25%	32.9M	1s
12050K	25%	31.8M	1s
12100K	25%	183M	1s
12150K	25%	68.3M	1s
12200K	25%	33.4M	1s
12250K	26%	27.1M	1s

12300K	26%	34.7M	1s
12350K	26%	35.1M	1s
12400K	26%	33.3M	1s
12450K	26%	56.8M	1s
12500K	26%	51.7M	1s
12550K	26%	35.1M	1s
12600K	26%	33.0M	1s
12650K	26%	36.6M	1s
12700K	26%	33.6M	1s
12750K	27%	36.4M	1s
12800K	27%	55.7M	1s
12850K	27%	51.1M	1s
12900K	27%	36.6M	1s
12950K	27%	55.1M	1s
13000K	27%	198M	1s
13050K	27%	38.3M	1s
13100K	27%	32.4M	1s
13150K	27%	35.3M	1s
13200K	28%	34.5M	1s
13250K	28%	30.7M	1s
13300K	28%	44.4M	1s
13350K	28%	79.7M	1s
13400K	28%	76.0M	1s
13450K	28%	31.3M	1s
13500K	28%	36.4M	1s
13550K	28%	33.5M	1s
13600K	28%	34.4M	1s
13650K	28%	64.4M	1s
13700K	29%	30.7M	1s
13750K	29%	34.0M	1s
13800K	29%	44.8M	1s
13850K	29%	53.5M	1s
13900K	29%	198M	1s
13950K	29%	44.2M	1s
14000K	29%	31.8M	1s
14050K	29%	30.4M	1s
14100K	29%	32.3M	1s
14150K	30%	34.3M	1s
14200K	30%	62.0M	1s
14250K	30%	31.1M	1s
14300K	30%	41.4M	1s
14350K	30%	34.2M	1s
14400K	30%	81.4M	1s
14450K	30%	27.6M	1s
14500K	30%	50.2M	1s
14550K	30%	76.6M	1s
14600K	30%	32.2M	1s
14650K	31%	28.3M	1s
14700K	31%	68.0M	1s
14750K	31%	159M	1s
14800K	31%	50.8M	1s

14850K	31%	9.56M	1s
14900K	31%	142M	1s
14950K	31%	147M	1s
15000K	31%	269M	1s
15050K	31%	99.1M	1s
15100K	32%	178M	1s
15150K	32%	178M	1s
15200K	32%	158M	1s
15250K	32%	159M	1s
15300K	32%	159M	1s
15350K	32%	148M	1s
15400K	32%	168M	1s
15450K	32%	147M	1s
15500K	32%	141M	1s
15550K	32%	175M	1s
15600K	33%	187M	1s
15650K	33%	165M	1s
15700K	33%	186M	1s
15750K	33%	174M	1s
15800K	33%	178M	1s
15850K	33%	151M	1s
15900K	33%	175M	1s
15950K	33%	185M	1s
16000K	33%	20.2M	1s
16050K	34%	158M	1s
16100K	34%	305M	1s
16150K	34%	78.4M	1s

*** WARNING: skipped 22420 bytes of output ***

30950K	65%	160M	0s
31000K	65%	148M	0s
31050K	65%	111M	0s
31100K	65%	143M	0s
31150K	65%	155M	0s
31200K	66%	126M	0s
31250K	66%	104M	0s
31300K	66%	151M	0s
31350K	66%	158M	0s
31400K	66%	153M	0s
31450K	66%	131M	0s
31500K	66%	141M	0s
31550K	66%	145M	0s
31600K	66%	146M	0s
31650K	67%	132M	0s
31700K	67%	153M	0s
31750K	67%	160M	0s
31800K	67%	126M	0s
31850K	67%	125M	0s
31900K	67%	157M	0s
31950K	67%	99.6M	0s

32000K	67%	155M	0s
32050K	67%	137M	0s
32100K	67%	128M	0s
32150K	68%	155M	0s
32200K	68%	106M	0s
32250K	68%	114M	0s
32300K	68%	163M	0s
32350K	68%	130M	0s
32400K	68%	182M	0s
32450K	68%	103M	0s
32500K	68%	139M	0s
32550K	68%	178M	0s
32600K	69%	144M	0s
32650K	69%	147M	0s
32700K	69%	164M	0s
32750K	69%	125M	0s
32800K	69%	141M	0s
32850K	69%	122M	0s
32900K	69%	143M	0s
32950K	69%	99.2M	0s
33000K	69%	129M	0s
33050K	69%	119M	0s
33100K	70%	145M	0s
33150K	70%	107M	0s
33200K	70%	141M	0s
33250K	70%	107M	0s
33300K	70%	148M	0s
33350K	70%	148M	0s
33400K	70%	120M	0s
33450K	70%	143M	0s
33500K	70%	114M	0s
33550K	71%	145M	0s
33600K	71%	161M	0s
33650K	71%	125M	0s
33700K	71%	154M	0s
33750K	71%	99.7M	0s
33800K	71%	136M	0s
33850K	71%	105M	0s
33900K	71%	119M	0s
33950K	71%	118M	0s
34000K	71%	127M	0s
34050K	72%	114M	0s
34100K	72%	136M	0s
34150K	72%	140M	0s
34200K	72%	125M	0s
34250K	72%	109M	0s
34300K	72%	132M	0s
34350K	72%	147M	0s
34400K	72%	91.6M	0s
34450K	72%	116M	0s
34500K	73%	106M	0s

34550K	73%	139M	0s
34600K	73%	121M	0s
34650K	73%	99.1M	0s
34700K	73%	142M	0s
34750K	73%	132M	0s
34800K	73%	134M	0s
34850K	73%	117M	0s
34900K	73%	149M	0s
34950K	74%	136M	0s
35000K	74%	109M	0s
35050K	74%	130M	0s
35100K	74%	96.2M	0s
35150K	74%	127M	0s
35200K	74%	184M	0s
35250K	74%	101M	0s
35300K	74%	111M	0s
35350K	74%	129M	0s
35400K	74%	134M	0s
35450K	75%	98.6M	0s
35500K	75%	149M	0s
35550K	75%	111M	0s
35600K	75%	149M	0s
35650K	75%	80.3M	0s
35700K	75%	150M	0s
35750K	75%	104M	0s
35800K	75%	148M	0s
35850K	75%	124M	0s
35900K	76%	149M	0s
35950K	76%	71.9M	0s
36000K	76%	71.4M	0s
36050K	76%	100M	0s
36100K	76%	82.3M	0s
36150K	76%	113M	0s
36200K	76%	142M	0s
36250K	76%	120M	0s
36300K	76%	76.8M	0s
36350K	76%	78.6M	0s
36400K	77%	146M	0s
36450K	77%	78.7M	0s
36500K	77%	147M	0s
36550K	77%	101M	0s
36600K	77%	150M	0s
36650K	77%	134M	0s
36700K	77%	146M	0s
36750K	77%	92.0M	0s
36800K	77%	106M	0s
36850K	78%	141M	0s
36900K	78%	269M	0s
36950K	78%	310M	0s
37000K	78%	265M	0s
37050K	78%	247M	0s

37100K	78%	275M	0s
37150K	78%	281M	0s
37200K	78%	267M	0s
37250K	78%	99.1M	0s
37300K	78%	264M	0s
37350K	79%	279M	0s
37400K	79%	279M	0s
37450K	79%	93.8M	0s
37500K	79%	296M	0s
37550K	79%	144M	0s
37600K	79%	228M	0s
37650K	79%	278M	0s
37700K	79%	280M	0s
37750K	79%	149M	0s
37800K	80%	277M	0s
37850K	80%	215M	0s
37900K	80%	248M	0s
37950K	80%	164M	0s
38000K	80%	302M	0s
38050K	80%	249M	0s
38100K	80%	275M	0s
38150K	80%	272M	0s
38200K	80%	301M	0s
38250K	80%	225M	0s
38300K	81%	309M	0s
38350K	81%	83.1M	0s
38400K	81%	140M	0s
38450K	81%	177M	0s
38500K	81%	281M	0s
38550K	81%	278M	0s
38600K	81%	275M	0s
38650K	81%	237M	0s
38700K	81%	289M	0s
38750K	82%	307M	0s
38800K	82%	286M	0s
38850K	82%	260M	0s
38900K	82%	82.4M	0s
38950K	82%	152M	0s
39000K	82%	151M	0s
39050K	82%	85.9M	0s
39100K	82%	118M	0s
39150K	82%	177M	0s
39200K	82%	162M	0s
39250K	83%	179M	0s
39300K	83%	175M	0s
39350K	83%	145M	0s
39400K	83%	166M	0s
39450K	83%	116M	0s
39500K	83%	165M	0s
39550K	83%	180M	0s
39600K	83%	161M	0s

39650K	83%	87.3M	0s
39700K	84%	157M	0s
39750K	84%	187M	0s
39800K	84%	178M	0s
39850K	84%	233M	0s
39900K	84%	281M	0s
39950K	84%	277M	0s
40000K	84%	299M	0s
40050K	84%	277M	0s
40100K	84%	174M	0s
40150K	84%	175M	0s
40200K	85%	136M	0s
40250K	85%	127M	0s
40300K	85%	177M	0s
40350K	85%	167M	0s
40400K	85%	171M	0s
40450K	85%	168M	0s
40500K	85%	168M	0s
40550K	85%	170M	0s
40600K	85%	172M	0s
40650K	86%	103M	0s
40700K	86%	167M	0s
40750K	86%	160M	0s
40800K	86%	160M	0s
40850K	86%	112M	0s
40900K	86%	137M	0s
40950K	86%	115M	0s
41000K	86%	38.8M	0s
41050K	86%	174M	0s
41100K	87%	195M	0s
41150K	87%	96.0M	0s
41200K	87%	86.9M	0s
41250K	87%	89.9M	0s
41300K	87%	102M	0s
41350K	87%	120M	0s
41400K	87%	66.3M	0s
41450K	87%	62.9M	0s
41500K	87%	130M	0s
41550K	87%	111M	0s
41600K	88%	117M	0s
41650K	88%	148M	0s
41700K	88%	94.5M	0s
41750K	88%	166M	0s
41800K	88%	154M	0s
41850K	88%	135M	0s
41900K	88%	166M	0s
41950K	88%	162M	0s
42000K	88%	89.7M	0s
42050K	89%	149M	0s
42100K	89%	161M	0s
42150K	89%	163M	0s

42200K	89%	160M	0s
42250K	89%	131M	0s
42300K	89%	161M	0s
42350K	89%	113M	0s
42400K	89%	162M	0s
42450K	89%	162M	0s
42500K	89%	158M	0s
42550K	90%	139M	0s
42600K	90%	161M	0s
42650K	90%	69.2M	0s
42700K	90%	156M	0s
42750K	90%	117M	0s
42800K	90%	163M	0s
42850K	90%	125M	0s
42900K	90%	153M	0s
42950K	90%	157M	0s
43000K	91%	144M	0s
43050K	91%	81.1M	0s
43100K	91%	62.4M	0s
43150K	91%	97.9M	0s
43200K	91%	144M	0s
43250K	91%	151M	0s
43300K	91%	68.8M	0s
43350K	91%	120M	0s
43400K	91%	127M	0s
43450K	91%	103M	0s
43500K	92%	139M	0s
43550K	92%	128M	0s
43600K	92%	101M	0s
43650K	92%	124M	0s
43700K	92%	128M	0s
43750K	92%	128M	0s
43800K	92%	145M	0s
43850K	92%	132M	0s
43900K	92%	129M	0s
43950K	93%	136M	0s
44000K	93%	61.8M	0s
44050K	93%	33.4M	0s
44100K	93%	147M	0s
44150K	93%	140M	0s
44200K	93%	137M	0s
44250K	93%	129M	0s
44300K	93%	152M	0s
44350K	93%	110M	0s
44400K	93%	144M	0s
44450K	94%	130M	0s
44500K	94%	132M	0s
44550K	94%	156M	0s
44600K	94%	159M	0s
44650K	94%	118M	0s
44700K	94%	145M	0s

44750K	94%	153M	0s
44800K	94%	164M	0s
44850K	94%	93.1M	0s
44900K	95%	137M	0s
44950K	95%	126M	0s
45000K	95%	153M	0s
45050K	95%	133M	0s
45100K	95%	137M	0s
45150K	95%	154M	0s
45200K	95%	157M	0s
45250K	95%	140M	0s
45300K	95%	142M	0s
45350K	95%	157M	0s
45400K	96%	149M	0s
45450K	96%	97.5M	0s
45500K	96%	123M	0s
45550K	96%	130M	0s
45600K	96%	175M	0s
45650K	96%	269M	0s
45700K	96%	302M	0s
45750K	96%	300M	0s
45800K	96%	280M	0s
45850K	97%	246M	0s
45900K	97%	245M	0s
45950K	97%	304M	0s
46000K	97%	164M	0s
46050K	97%	199M	0s
46100K	97%	170M	0s
46150K	97%	198M	0s
46200K	97%	186M	0s
46250K	97%	146M	0s
46300K	97%	175M	0s
46350K	98%	216M	0s
46400K	98%	213M	0s
46450K	98%	266M	0s
46500K	98%	312M	0s
46550K	98%	272M	0s
46600K	98%	310M	0s
46650K	98%	241M	0s
46700K	98%	262M	0s
46750K	98%	304M	0s
46800K	99%	306M	0s
46850K	99%	258M	0s
46900K	99%	133M	0s
46950K	99%	254M	0s
47000K	99%	299M	0s
47050K	99%	255M	0s
47100K	99%	308M	0s
47150K	99%	280M	0s
47200K	99%	305M	0s
47250K	100%	272M=0.6s	

2020-05-31 14:09:26 (72.6 MB/s) - 'title.crew.tsv.gz' saved [48431137/48431137]

/bin/bash: line 1: fg: no job control

```
In [10]: title_crew = spark.read.option("sep", "\t").csv('file:/databricks/driver/titl
e.crew.tsv', header=True, inferSchema = True)
title_crew.cache()
title_crew.show(3)
```

```
+-----+-----+-----+
      tconst|directors|writers|
+-----+-----+-----+
tt0000001|nm0005690|      \N|
tt0000002|nm0721526|      \N|
tt0000003|nm0721526|      \N|
+-----+-----+-----+
only showing top 3 rows
```

```
In [11]: %sh wget https://datasets.imdbws.com/title.crew.tsv.gz  
%sh  
gunzip title.crew.tsv.gz
```



```
--2020-05-31 14:09:40-- https://datasets.imdbws.com/title.crew.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.26, 13.224.13.32,
13.224.13.37, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.26|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 48431137 (46M) [binary/octet-stream]
Saving to: 'title.crew.tsv.gz'
```

0K	0%	5.38M	9s
50K	0%	11.3M	6s
100K	0%	17.7M	5s
150K	0%	23.3M	4s
200K	0%	18.3M	4s
250K	0%	32.3M	4s
300K	0%	34.2M	3s
350K	0%	31.9M	3s
400K	0%	32.1M	3s
450K	1%	36.2M	3s
500K	1%	33.1M	3s
550K	1%	32.0M	2s
600K	1%	28.2M	2s
650K	1%	34.1M	2s
700K	1%	31.8M	2s
750K	1%	30.3M	2s
800K	1%	34.4M	2s
850K	1%	34.8M	2s
900K	2%	38.9M	2s
950K	2%	36.9M	2s
1000K	2%	36.6M	2s
1050K	2%	39.4M	2s
1100K	2%	33.6M	2s
1150K	2%	32.8M	2s
1200K	2%	33.7M	2s
1250K	2%	34.6M	2s
1300K	2%	33.0M	2s
1350K	2%	36.3M	2s
1400K	3%	31.6M	2s
1450K	3%	34.1M	2s
1500K	3%	36.8M	2s
1550K	3%	32.0M	2s
1600K	3%	37.3M	2s
1650K	3%	33.9M	2s
1700K	3%	29.6M	2s
1750K	3%	27.4M	2s
1800K	3%	24.6M	2s
1850K	4%	27.2M	2s
1900K	4%	172M	2s
1950K	4%	27.2M	2s
2000K	4%	43.1M	2s
2050K	4%	45.0M	2s

2100K	4%	35.8M	2s
2150K	4%	42.4M	2s
2200K	4%	47.7M	2s
2250K	4%	40.9M	2s
2300K	4%	44.4M	2s
2350K	5%	189M	2s
2400K	5%	46.4M	1s
2450K	5%	36.0M	1s
2500K	5%	36.1M	1s
2550K	5%	38.0M	1s
2600K	5%	53.6M	1s
2650K	5%	43.2M	1s
2700K	5%	70.9M	1s
2750K	5%	31.2M	1s
2800K	6%	33.5M	1s
2850K	6%	61.3M	1s
2900K	6%	40.3M	1s
2950K	6%	45.7M	1s
3000K	6%	30.9M	1s
3050K	6%	52.7M	1s
3100K	6%	46.0M	1s
3150K	6%	38.2M	1s
3200K	6%	55.5M	1s
3250K	6%	35.3M	1s
3300K	7%	63.4M	1s
3350K	7%	38.1M	1s
3400K	7%	36.8M	1s
3450K	7%	81.5M	1s
3500K	7%	35.1M	1s
3550K	7%	97.7M	1s
3600K	7%	45.6M	1s
3650K	7%	39.0M	1s
3700K	7%	35.3M	1s
3750K	8%	36.4M	1s
3800K	8%	111M	1s
3850K	8%	65.2M	1s
3900K	8%	36.6M	1s
3950K	8%	33.6M	1s
4000K	8%	32.9M	1s
4050K	8%	99.6M	1s
4100K	8%	201M	1s
4150K	8%	46.7M	1s
4200K	8%	32.7M	1s
4250K	9%	38.0M	1s
4300K	9%	37.1M	1s
4350K	9%	47.4M	1s
4400K	9%	35.8M	1s
4450K	9%	210M	1s
4500K	9%	37.1M	1s
4550K	9%	36.0M	1s
4600K	9%	36.8M	1s

4650K	9%	45.7M	1s
4700K	10%	103M	1s
4750K	10%	32.7M	1s
4800K	10%	30.5M	1s
4850K	10%	33.3M	1s
4900K	10%	50.3M	1s
4950K	10%	93.7M	1s
5000K	10%	29.7M	1s
5050K	10%	38.0M	1s
5100K	10%	53.6M	1s
5150K	10%	102M	1s
5200K	11%	36.1M	1s
5250K	11%	41.3M	1s
5300K	11%	83.7M	1s
5350K	11%	36.1M	1s
5400K	11%	41.7M	1s
5450K	11%	45.2M	1s
5500K	11%	40.5M	1s
5550K	11%	34.6M	1s
5600K	11%	57.9M	1s
5650K	12%	37.3M	1s
5700K	12%	91.1M	1s
5750K	12%	43.7M	1s
5800K	12%	33.5M	1s
5850K	12%	65.1M	1s
5900K	12%	44.0M	1s
5950K	12%	35.4M	1s
6000K	12%	143M	1s
6050K	12%	35.6M	1s
6100K	13%	36.2M	1s
6150K	13%	73.1M	1s
6200K	13%	31.9M	1s
6250K	13%	48.8M	1s
6300K	13%	203M	1s
6350K	13%	38.9M	1s
6400K	13%	34.8M	1s
6450K	13%	29.3M	1s
6500K	13%	40.4M	1s
6550K	13%	82.0M	1s
6600K	14%	44.2M	1s
6650K	14%	37.5M	1s
6700K	14%	47.7M	1s
6750K	14%	34.5M	1s
6800K	14%	92.4M	1s
6850K	14%	37.3M	1s
6900K	14%	48.6M	1s
6950K	14%	215M	1s
7000K	14%	34.0M	1s
7050K	15%	32.8M	1s
7100K	15%	37.6M	1s
7150K	15%	72.0M	1s

7200K	15%	45.8M	1s
7250K	15%	37.9M	1s
7300K	15%	35.8M	1s
7350K	15%	86.5M	1s
7400K	15%	35.1M	1s
7450K	15%	49.4M	1s
7500K	15%	38.9M	1s
7550K	16%	74.4M	1s
7600K	16%	41.8M	1s
7650K	16%	37.7M	1s
7700K	16%	39.4M	1s
7750K	16%	57.2M	1s
7800K	16%	55.5M	1s
7850K	16%	210M	1s
7900K	16%	43.5M	1s
7950K	16%	30.9M	1s
8000K	17%	35.6M	1s
8050K	17%	34.0M	1s
8100K	17%	39.7M	1s
8150K	17%	50.7M	1s
8200K	17%	50.8M	1s
8250K	17%	38.7M	1s
8300K	17%	50.6M	1s
8350K	17%	170M	1s
8400K	17%	49.3M	1s
8450K	17%	38.0M	1s
8500K	18%	32.4M	1s
8550K	18%	37.8M	1s
8600K	18%	90.3M	1s
8650K	18%	57.7M	1s
8700K	18%	35.6M	1s
8750K	18%	32.6M	1s
8800K	18%	36.1M	1s
8850K	18%	117M	1s
8900K	18%	207M	1s
8950K	19%	48.7M	1s
9000K	19%	32.4M	1s
9050K	19%	37.8M	1s
9100K	19%	37.3M	1s
9150K	19%	36.5M	1s
9200K	19%	36.4M	1s
9250K	19%	68.0M	1s
9300K	19%	37.3M	1s
9350K	19%	49.3M	1s
9400K	19%	34.4M	1s
9450K	20%	58.4M	1s
9500K	20%	205M	1s
9550K	20%	84.9M	1s
9600K	20%	30.0M	1s
9650K	20%	35.0M	1s
9700K	20%	35.6M	1s

9750K	20%	46.6M	1s
9800K	20%	35.0M	1s
9850K	20%	117M	1s
9900K	21%	207M	1s
9950K	21%	22.6M	1s
10000K	21%	32.2M	1s
10050K	21%	35.1M	1s
10100K	21%	34.8M	1s
10150K	21%	36.3M	1s
10200K	21%	70.1M	1s
10250K	21%	207M	1s
10300K	21%	190M	1s
10350K	21%	39.0M	1s
10400K	22%	34.9M	1s
10450K	22%	38.7M	1s
10500K	22%	39.2M	1s
10550K	22%	37.7M	1s
10600K	22%	35.0M	1s
10650K	22%	35.1M	1s
10700K	22%	37.1M	1s
10750K	22%	40.8M	1s
10800K	22%	192M	1s
10850K	23%	205M	1s
10900K	23%	67.6M	1s
10950K	23%	38.8M	1s
11000K	23%	33.2M	1s
11050K	23%	32.7M	1s
11100K	23%	36.8M	1s
11150K	23%	31.1M	1s
11200K	23%	51.6M	1s
11250K	23%	49.5M	1s
11300K	23%	35.3M	1s
11350K	24%	60.4M	1s
11400K	24%	187M	1s
11450K	24%	88.4M	1s
11500K	24%	34.8M	1s
11550K	24%	32.5M	1s
11600K	24%	30.7M	1s
11650K	24%	35.9M	1s
11700K	24%	32.0M	1s
11750K	24%	36.9M	1s
11800K	25%	42.3M	1s
11850K	25%	44.7M	1s
11900K	25%	39.5M	1s
11950K	25%	72.4M	1s
12000K	25%	113M	1s
12050K	25%	29.9M	1s
12100K	25%	32.8M	1s
12150K	25%	31.3M	1s
12200K	25%	34.0M	1s
12250K	26%	111M	1s

12300K	26%	18.7M	1s
12350K	26%	31.6M	1s
12400K	26%	38.8M	1s
12450K	26%	59.1M	1s
12500K	26%	203M	1s
12550K	26%	210M	1s
12600K	26%	38.5M	1s
12650K	26%	33.1M	1s
12700K	26%	36.6M	1s
12750K	27%	33.3M	1s
12800K	27%	31.2M	1s
12850K	27%	34.2M	1s
12900K	27%	37.1M	1s
12950K	27%	57.3M	1s
13000K	27%	82.8M	1s
13050K	27%	33.2M	1s
13100K	27%	44.6M	1s
13150K	27%	46.1M	1s
13200K	28%	52.6M	1s
13250K	28%	7.55M	1s
13300K	28%	89.0M	1s
13350K	28%	34.3M	1s
13400K	28%	31.2M	1s
13450K	28%	48.4M	1s
13500K	28%	69.7M	1s
13550K	28%	33.7M	1s
13600K	28%	36.5M	1s
13650K	28%	46.6M	1s
13700K	29%	49.2M	1s
13750K	29%	44.6M	1s
13800K	29%	51.2M	1s
13850K	29%	213M	1s
13900K	29%	66.4M	1s
13950K	29%	32.8M	1s
14000K	29%	33.3M	1s
14050K	29%	36.0M	1s
14100K	29%	144M	1s
14150K	30%	178M	1s
14200K	30%	128M	1s
14250K	30%	191M	1s
14300K	30%	317M	1s
14350K	30%	264M	1s
14400K	30%	127M	1s
14450K	30%	141M	1s
14500K	30%	171M	1s
14550K	30%	173M	1s
14600K	30%	137M	1s
14650K	31%	313M	1s
14700K	31%	296M	1s
14750K	31%	98.4M	1s
14800K	31%	24.3M	1s

14850K	31%	50.3M	1s
14900K	31%	161M	1s
14950K	31%	134M	1s
15000K	31%	79.5M	1s
15050K	31%	79.1M	1s
15100K	32%	78.1M	1s
15150K	32%	167M	1s
15200K	32%	66.6M	1s
15250K	32%	81.7M	1s
15300K	32%	84.2M	1s
15350K	32%	86.9M	1s
15400K	32%	125M	1s
15450K	32%	85.9M	1s
15500K	32%	81.4M	1s
15550K	32%	84.8M	1s
15600K	33%	81.8M	1s
15650K	33%	139M	1s
15700K	33%	96.5M	1s
15750K	33%	79.9M	1s
15800K	33%	78.3M	1s
15850K	33%	156M	1s
15900K	33%	85.3M	1s
15950K	33%	79.6M	1s
16000K	33%	76.8M	1s
16050K	34%	88.4M	1s
16100K	34%	238M	1s
16150K	34%	67.1M	1s

*** WARNING: skipped 22496 bytes of output ***

31000K	65%	122M	0s
31050K	65%	151M	0s
31100K	65%	202M	0s
31150K	65%	296M	0s
31200K	66%	302M	0s
31250K	66%	295M	0s
31300K	66%	134M	0s
31350K	66%	172M	0s
31400K	66%	36.9M	0s
31450K	66%	86.3M	0s
31500K	66%	157M	0s
31550K	66%	137M	0s
31600K	66%	140M	0s
31650K	67%	59.9M	0s
31700K	67%	78.8M	0s
31750K	67%	84.4M	0s
31800K	67%	138M	0s
31850K	67%	151M	0s
31900K	67%	83.3M	0s
31950K	67%	68.3M	0s
32000K	67%	223M	0s

32050K	67%	283M	0s
32100K	67%	317M	0s
32150K	68%	297M	0s
32200K	68%	251M	0s
32250K	68%	277M	0s
32300K	68%	329M	0s
32350K	68%	223M	0s
32400K	68%	78.0M	0s
32450K	68%	111M	0s
32500K	68%	147M	0s
32550K	68%	152M	0s
32600K	69%	137M	0s
32650K	69%	166M	0s
32700K	69%	158M	0s
32750K	69%	136M	0s
32800K	69%	148M	0s
32850K	69%	317M	0s
32900K	69%	331M	0s
32950K	69%	69.8M	0s
33000K	69%	137M	0s
33050K	69%	153M	0s
33100K	70%	156M	0s
33150K	70%	68.0M	0s
33200K	70%	105M	0s
33250K	70%	192M	0s
33300K	70%	328M	0s
33350K	70%	299M	0s
33400K	70%	99.1M	0s
33450K	70%	144M	0s
33500K	70%	92.7M	0s
33550K	71%	140M	0s
33600K	71%	139M	0s
33650K	71%	166M	0s
33700K	71%	160M	0s
33750K	71%	80.0M	0s
33800K	71%	62.4M	0s
33850K	71%	59.3M	0s
33900K	71%	72.3M	0s
33950K	71%	76.7M	0s
34000K	71%	72.5M	0s
34050K	72%	184M	0s
34100K	72%	321M	0s
34150K	72%	330M	0s
34200K	72%	262M	0s
34250K	72%	326M	0s
34300K	72%	288M	0s
34350K	72%	287M	0s
34400K	72%	217M	0s
34450K	72%	315M	0s
34500K	73%	282M	0s
34550K	73%	265M	0s

34600K	73%	49.5M	0s
34650K	73%	87.6M	0s
34700K	73%	153M	0s
34750K	73%	140M	0s
34800K	73%	103M	0s
34850K	73%	155M	0s
34900K	73%	146M	0s
34950K	74%	157M	0s
35000K	74%	131M	0s
35050K	74%	158M	0s
35100K	74%	149M	0s
35150K	74%	137M	0s
35200K	74%	138M	0s
35250K	74%	105M	0s
35300K	74%	65.7M	0s
35350K	74%	66.7M	0s
35400K	74%	95.5M	0s
35450K	75%	166M	0s
35500K	75%	316M	0s
35550K	75%	258M	0s
35600K	75%	297M	0s
35650K	75%	124M	0s
35700K	75%	285M	0s
35750K	75%	326M	0s
35800K	75%	259M	0s
35850K	75%	323M	0s
35900K	76%	331M	0s
35950K	76%	266M	0s
36000K	76%	288M	0s
36050K	76%	298M	0s
36100K	76%	320M	0s
36150K	76%	284M	0s
36200K	76%	298M	0s
36250K	76%	321M	0s
36300K	76%	283M	0s
36350K	76%	289M	0s
36400K	77%	272M	0s
36450K	77%	328M	0s
36500K	77%	277M	0s
36550K	77%	34.3M	0s
36600K	77%	55.7M	0s
36650K	77%	154M	0s
36700K	77%	149M	0s
36750K	77%	72.8M	0s
36800K	77%	118M	0s
36850K	78%	140M	0s
36900K	78%	152M	0s
36950K	78%	195M	0s
37000K	78%	285M	0s
37050K	78%	300M	0s
37100K	78%	321M	0s

37150K	78%	260M	0s
37200K	78%	297M	0s
37250K	78%	327M	0s
37300K	78%	290M	0s
37350K	79%	305M	0s
37400K	79%	140M	0s
37450K	79%	123M	0s
37500K	79%	179M	0s
37550K	79%	124M	0s
37600K	79%	165M	0s
37650K	79%	44.3M	0s
37700K	79%	101M	0s
37750K	79%	134M	0s
37800K	80%	135M	0s
37850K	80%	148M	0s
37900K	80%	298M	0s
37950K	80%	280M	0s
38000K	80%	294M	0s
38050K	80%	282M	0s
38100K	80%	336M	0s
38150K	80%	324M	0s
38200K	80%	264M	0s
38250K	80%	287M	0s
38300K	81%	329M	0s
38350K	81%	56.8M	0s
38400K	81%	70.3M	0s
38450K	81%	69.9M	0s
38500K	81%	66.7M	0s
38550K	81%	68.8M	0s
38600K	81%	114M	0s
38650K	81%	138M	0s
38700K	81%	226M	0s
38750K	82%	201M	0s
38800K	82%	196M	0s
38850K	82%	223M	0s
38900K	82%	216M	0s
38950K	82%	218M	0s
39000K	82%	196M	0s
39050K	82%	212M	0s
39100K	82%	215M	0s
39150K	82%	91.0M	0s
39200K	82%	154M	0s
39250K	83%	129M	0s
39300K	83%	143M	0s
39350K	83%	39.8M	0s
39400K	83%	135M	0s
39450K	83%	148M	0s
39500K	83%	157M	0s
39550K	83%	133M	0s
39600K	83%	145M	0s
39650K	83%	150M	0s

39700K	84%	159M	0s
39750K	84%	159M	0s
39800K	84%	128M	0s
39850K	84%	152M	0s
39900K	84%	157M	0s
39950K	84%	57.3M	0s
40000K	84%	67.8M	0s
40050K	84%	101M	0s
40100K	84%	127M	0s
40150K	84%	194M	0s
40200K	85%	181M	0s
40250K	85%	146M	0s
40300K	85%	134M	0s
40350K	85%	133M	0s
40400K	85%	140M	0s
40450K	85%	149M	0s
40500K	85%	154M	0s
40550K	85%	163M	0s
40600K	85%	142M	0s
40650K	86%	152M	0s
40700K	86%	150M	0s
40750K	86%	73.4M	0s
40800K	86%	68.1M	0s
40850K	86%	71.3M	0s
40900K	86%	132M	0s
40950K	86%	323M	0s
41000K	86%	137M	0s
41050K	86%	263M	0s
41100K	87%	302M	0s
41150K	87%	250M	0s
41200K	87%	306M	0s
41250K	87%	279M	0s
41300K	87%	295M	0s
41350K	87%	248M	0s
41400K	87%	304M	0s
41450K	87%	306M	0s
41500K	87%	266M	0s
41550K	87%	64.6M	0s
41600K	88%	164M	0s
41650K	88%	301M	0s
41700K	88%	272M	0s
41750K	88%	241M	0s
41800K	88%	114M	0s
41850K	88%	171M	0s
41900K	88%	122M	0s
41950K	88%	176M	0s
42000K	88%	130M	0s
42050K	89%	158M	0s
42100K	89%	179M	0s
42150K	89%	53.0M	0s
42200K	89%	302M	0s

42250K	89%	306M	0s
42300K	89%	313M	0s
42350K	89%	273M	0s
42400K	89%	266M	0s
42450K	89%	291M	0s
42500K	89%	283M	0s
42550K	90%	254M	0s
42600K	90%	309M	0s
42650K	90%	299M	0s
42700K	90%	267M	0s
42750K	90%	271M	0s
42800K	90%	276M	0s
42850K	90%	309M	0s
42900K	90%	304M	0s
42950K	90%	249M	0s
43000K	91%	290M	0s
43050K	91%	46.6M	0s
43100K	91%	72.3M	0s
43150K	91%	67.3M	0s
43200K	91%	56.3M	0s
43250K	91%	70.4M	0s
43300K	91%	72.6M	0s
43350K	91%	123M	0s
43400K	91%	159M	0s
43450K	91%	162M	0s
43500K	92%	156M	0s
43550K	92%	144M	0s
43600K	92%	74.3M	0s
43650K	92%	129M	0s
43700K	92%	109M	0s
43750K	92%	103M	0s
43800K	92%	126M	0s
43850K	92%	189M	0s
43900K	92%	117M	0s
43950K	93%	163M	0s
44000K	93%	129M	0s
44050K	93%	246M	0s
44100K	93%	276M	0s
44150K	93%	231M	0s
44200K	93%	294M	0s
44250K	93%	300M	0s
44300K	93%	302M	0s
44350K	93%	251M	0s
44400K	93%	232M	0s
44450K	94%	267M	0s
44500K	94%	291M	0s
44550K	94%	252M	0s
44600K	94%	306M	0s
44650K	94%	45.7M	0s
44700K	94%	71.8M	0s
44750K	94%	145M	0s

44800K	94%	151M	0s
44850K	94%	131M	0s
44900K	95%	271M	0s
44950K	95%	242M	0s
45000K	95%	156M	0s
45050K	95%	261M	0s
45100K	95%	212M	0s
45150K	95%	170M	0s
45200K	95%	192M	0s
45250K	95%	232M	0s
45300K	95%	288M	0s
45350K	95%	250M	0s
45400K	96%	275M	0s
45450K	96%	299M	0s
45500K	96%	118M	0s
45550K	96%	103M	0s
45600K	96%	179M	0s
45650K	96%	110M	0s
45700K	96%	169M	0s
45750K	96%	104M	0s
45800K	96%	38.8M	0s
45850K	97%	118M	0s
45900K	97%	158M	0s
45950K	97%	147M	0s
46000K	97%	151M	0s
46050K	97%	154M	0s
46100K	97%	155M	0s
46150K	97%	147M	0s
46200K	97%	267M	0s
46250K	97%	307M	0s
46300K	97%	276M	0s
46350K	98%	232M	0s
46400K	98%	106M	0s
46450K	98%	154M	0s
46500K	98%	196M	0s
46550K	98%	246M	0s
46600K	98%	300M	0s
46650K	98%	275M	0s
46700K	98%	262M	0s
46750K	98%	277M	0s
46800K	99%	268M	0s
46850K	99%	295M	0s
46900K	99%	310M	0s
46950K	99%	208M	0s
47000K	99%	254M	0s
47050K	99%	301M	0s
47100K	99%	283M	0s
47150K	99%	271M	0s
47200K	99%	297M	0s
47250K	100%	252M	=0.6s

2020-05-31 14:09:41 (78.9 MB/s) - 'title.crew.tsv.gz' saved [48431137/48431137]

/bin/bash: line 1: fg: no job control

gzip: title.crew.tsv already exists; not overwritten

```
In [12]: title_crew = spark.read.option("sep", "\t").csv('file:/databricks/driver/titl
e.crew.tsv', header=True, inferSchema = True)
title_crew.cache()
title_crew.show(3)
```

```
+-----+-----+-----+
      tconst|directors|writers|
```

```
+-----+-----+-----+
```

```
tt0000001|nm0005690|      \N|
```

```
tt0000002|nm0721526|      \N|
```

```
tt0000003|nm0721526|      \N|
```

```
+-----+-----+-----+
```

only showing top 3 rows

```
In [13]: %sh wget https://datasets.imdbws.com/title.episode.tsv.gz  
%sh  
gunzip title.episode.tsv.gz
```

```
--2020-05-31 14:09:51-- https://datasets.imdbws.com/title.episode.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.54, 13.224.13.26,
13.224.13.32, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.54|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 26597530 (25M) [binary/octet-stream]
Saving to: 'title.episode.tsv.gz'
```

0K	0%	3.87M	7s
50K	0%	8.53M	5s
100K	0%	14.1M	4s
150K	0%	14.2M	3s
200K	0%	19.5M	3s
250K	1%	19.7M	3s
300K	1%	34.1M	2s
350K	1%	22.6M	2s
400K	1%	28.5M	2s
450K	1%	45.2M	2s
500K	2%	36.2M	2s
550K	2%	54.0M	2s
600K	2%	42.9M	2s
650K	2%	66.1M	1s
700K	2%	60.5M	1s
750K	3%	42.9M	1s
800K	3%	87.8M	1s
850K	3%	60.1M	1s
900K	3%	66.3M	1s
950K	3%	129M	1s
1000K	4%	61.6M	1s
1050K	4%	115M	1s
1100K	4%	62.7M	1s
1150K	4%	87.8M	1s
1200K	4%	201M	1s
1250K	5%	68.8M	1s
1300K	5%	108M	1s
1350K	5%	90.7M	1s
1400K	5%	154M	1s
1450K	5%	128M	1s
1500K	5%	86.6M	1s
1550K	6%	94.2M	1s
1600K	6%	101M	1s
1650K	6%	114M	1s
1700K	6%	182M	1s
1750K	6%	158M	1s
1800K	7%	118M	1s
1850K	7%	114M	1s
1900K	7%	120M	1s
1950K	7%	115M	1s
2000K	7%	137M	1s
2050K	8%	120M	1s

2100K	8%	110M	1s
2150K	8%	126M	1s
2200K	8%	135M	1s
2250K	8%	120M	1s
2300K	9%	123M	1s
2350K	9%	130M	1s
2400K	9%	193M	1s
2450K	9%	193M	1s
2500K	9%	141M	1s
2550K	10%	157M	1s
2600K	10%	188M	1s
2650K	10%	135M	1s
2700K	10%	189M	1s
2750K	10%	165M	0s
2800K	10%	300M	0s
2850K	11%	179M	0s
2900K	11%	225M	0s
2950K	11%	250M	0s
3000K	11%	218M	0s
3050K	11%	301M	0s
3100K	12%	203M	0s
3150K	12%	212M	0s
3200K	12%	201M	0s
3250K	12%	206M	0s
3300K	12%	177M	0s
3350K	13%	179M	0s
3400K	13%	203M	0s
3450K	13%	183M	0s
3500K	13%	175M	0s
3550K	13%	156M	0s
3600K	14%	205M	0s
3650K	14%	15.5M	0s
3700K	14%	37.0M	0s
3750K	14%	32.9M	0s
3800K	14%	32.5M	0s
3850K	15%	105M	0s
3900K	15%	108M	0s
3950K	15%	140M	0s
4000K	15%	145M	0s
4050K	15%	107M	0s
4100K	15%	121M	0s
4150K	16%	105M	0s
4200K	16%	124M	0s
4250K	16%	137M	0s
4300K	16%	147M	0s
4350K	16%	123M	0s
4400K	17%	159M	0s
4450K	17%	101M	0s
4500K	17%	93.2M	0s
4550K	17%	92.9M	0s
4600K	17%	155M	0s

4650K	18%	159M	0s
4700K	18%	115M	0s
4750K	18%	143M	0s
4800K	18%	146M	0s
4850K	18%	129M	0s
4900K	19%	127M	0s
4950K	19%	162M	0s
5000K	19%	158M	0s
5050K	19%	127M	0s
5100K	19%	130M	0s
5150K	20%	123M	0s
5200K	20%	135M	0s
5250K	20%	157M	0s
5300K	20%	136M	0s
5350K	20%	117M	0s
5400K	20%	148M	0s
5450K	21%	136M	0s
5500K	21%	125M	0s
5550K	21%	152M	0s
5600K	21%	156M	0s
5650K	21%	160M	0s
5700K	22%	121M	0s
5750K	22%	169M	0s
5800K	22%	160M	0s
5850K	22%	147M	0s
5900K	22%	133M	0s
5950K	23%	124M	0s
6000K	23%	131M	0s
6050K	23%	159M	0s
6100K	23%	137M	0s
6150K	23%	162M	0s
6200K	24%	148M	0s
6250K	24%	163M	0s
6300K	24%	156M	0s
6350K	24%	163M	0s
6400K	24%	169M	0s
6450K	25%	158M	0s
6500K	25%	92.3M	0s
6550K	25%	135M	0s
6600K	25%	159M	0s
6650K	25%	182M	0s
6700K	25%	98.3M	0s
6750K	26%	134M	0s
6800K	26%	158M	0s
6850K	26%	130M	0s
6900K	26%	124M	0s
6950K	26%	20.2M	0s
7000K	27%	31.6M	0s
7050K	27%	19.9M	0s
7100K	27%	41.4M	0s
7150K	27%	130M	0s

7200K	27%	144M	0s
7250K	28%	78.8M	0s
7300K	28%	30.6M	0s
7350K	28%	153M	0s
7400K	28%	158M	0s
7450K	28%	146M	0s
7500K	29%	143M	0s
7550K	29%	150M	0s
7600K	29%	145M	0s
7650K	29%	149M	0s
7700K	29%	137M	0s
7750K	30%	119M	0s
7800K	30%	161M	0s
7850K	30%	155M	0s
7900K	30%	144M	0s
7950K	30%	160M	0s
8000K	30%	154M	0s
8050K	31%	140M	0s
8100K	31%	119M	0s
8150K	31%	162M	0s
8200K	31%	157M	0s
8250K	31%	157M	0s
8300K	32%	135M	0s
8350K	32%	192M	0s
8400K	32%	278M	0s
8450K	32%	296M	0s
8500K	32%	236M	0s
8550K	33%	267M	0s
8600K	33%	313M	0s
8650K	33%	281M	0s
8700K	33%	127M	0s
8750K	33%	215M	0s
8800K	34%	194M	0s
8850K	34%	314M	0s
8900K	34%	219M	0s
8950K	34%	286M	0s
9000K	34%	312M	0s
9050K	35%	281M	0s
9100K	35%	283M	0s
9150K	35%	317M	0s
9200K	35%	281M	0s
9250K	35%	288M	0s
9300K	35%	237M	0s
9350K	36%	134M	0s
9400K	36%	302M	0s
9450K	36%	175M	0s
9500K	36%	240M	0s
9550K	36%	286M	0s
9600K	37%	274M	0s
9650K	37%	298M	0s
9700K	37%	10.3M	0s

9750K	37%	86.0M	0s
9800K	37%	121M	0s
9850K	38%	90.1M	0s
9900K	38%	81.5M	0s
9950K	38%	152M	0s
10000K	38%	154M	0s
10050K	38%	174M	0s
10100K	39%	122M	0s
10150K	39%	156M	0s
10200K	39%	284M	0s
10250K	39%	278M	0s
10300K	39%	277M	0s
10350K	40%	306M	0s
10400K	40%	276M	0s
10450K	40%	11.2M	0s
10500K	40%	65.1M	0s
10550K	40%	85.1M	0s
10600K	41%	84.2M	0s
10650K	41%	118M	0s
10700K	41%	82.6M	0s
10750K	41%	91.2M	0s
10800K	41%	85.0M	0s
10850K	41%	94.6M	0s
10900K	42%	83.9M	0s
10950K	42%	88.6M	0s
11000K	42%	115M	0s
11050K	42%	119M	0s
11100K	42%	274M	0s
11150K	43%	299M	0s
11200K	43%	269M	0s
11250K	43%	268M	0s
11300K	43%	244M	0s
11350K	43%	272M	0s
11400K	44%	297M	0s
11450K	44%	307M	0s
11500K	44%	250M	0s
11550K	44%	117M	0s
11600K	44%	145M	0s
11650K	45%	53.5M	0s
11700K	45%	84.7M	0s
11750K	45%	160M	0s
11800K	45%	156M	0s
11850K	45%	161M	0s
11900K	46%	101M	0s
11950K	46%	160M	0s
12000K	46%	154M	0s
12050K	46%	161M	0s
12100K	46%	53.3M	0s
12150K	46%	95.8M	0s
12200K	47%	166M	0s
12250K	47%	196M	0s

12300K	47%	153M	0s
12350K	47%	266M	0s
12400K	47%	297M	0s
12450K	48%	272M	0s
12500K	48%	280M	0s
12550K	48%	248M	0s
12600K	48%	275M	0s
12650K	48%	237M	0s
12700K	49%	293M	0s
12750K	49%	224M	0s
12800K	49%	306M	0s
12850K	49%	48.9M	0s
12900K	49%	180M	0s
12950K	50%	150M	0s
13000K	50%	182M	0s
13050K	50%	182M	0s
13100K	50%	178M	0s
13150K	50%	162M	0s
13200K	51%	175M	0s
13250K	51%	302M	0s
13300K	51%	120M	0s
13350K	51%	145M	0s
13400K	51%	57.6M	0s
13450K	51%	86.7M	0s
13500K	52%	156M	0s
13550K	52%	141M	0s
13600K	52%	115M	0s
13650K	52%	111M	0s
13700K	52%	137M	0s
13750K	53%	134M	0s
13800K	53%	154M	0s
13850K	53%	158M	0s
13900K	53%	163M	0s
13950K	53%	89.3M	0s
14000K	54%	94.9M	0s
14050K	54%	127M	0s
14100K	54%	99.2M	0s
14150K	54%	150M	0s
14200K	54%	179M	0s
14250K	55%	180M	0s
14300K	55%	165M	0s
14350K	55%	143M	0s
14400K	55%	57.3M	0s
14450K	55%	94.3M	0s
14500K	56%	84.5M	0s
14550K	56%	225M	0s
14600K	56%	307M	0s
14650K	56%	305M	0s
14700K	56%	308M	0s
14750K	56%	120M	0s
14800K	57%	250M	0s

14850K	57%	253M	0s
14900K	57%	315M	0s
14950K	57%	251M	0s
15000K	57%	301M	0s
15050K	58%	211M	0s
15100K	58%	71.1M	0s
15150K	58%	147M	0s
15200K	58%	89.7M	0s
15250K	58%	87.8M	0s
15300K	59%	161M	0s
15350K	59%	107M	0s
15400K	59%	184M	0s
15450K	59%	154M	0s
15500K	59%	174M	0s
15550K	60%	87.7M	0s
15600K	60%	105M	0s
15650K	60%	156M	0s
15700K	60%	160M	0s
15750K	60%	136M	0s
15800K	61%	268M	0s
15850K	61%	300M	0s
15900K	61%	303M	0s
15950K	61%	271M	0s
16000K	61%	298M	0s
16050K	61%	266M	0s
16100K	62%	277M	0s
16150K	62%	111M	0s
16200K	62%	297M	0s
16250K	62%	135M	0s
16300K	62%	133M	0s
16350K	63%	272M	0s
16400K	63%	315M	0s
16450K	63%	299M	0s
16500K	63%	271M	0s
16550K	63%	225M	0s
16600K	64%	274M	0s
16650K	64%	313M	0s
16700K	64%	295M	0s
16750K	64%	112M	0s
16800K	64%	153M	0s
16850K	65%	155M	0s
16900K	65%	192M	0s
16950K	65%	146M	0s
17000K	65%	115M	0s
17050K	65%	76.3M	0s
17100K	66%	48.4M	0s
17150K	66%	158M	0s
17200K	66%	190M	0s
17250K	66%	187M	0s
17300K	66%	212M	0s
17350K	66%	174M	0s

17400K	67%	155M	0s
17450K	67%	208M	0s
17500K	67%	217M	0s
17550K	67%	92.6M	0s
17600K	67%	94.4M	0s
17650K	68%	141M	0s
17700K	68%	99.3M	0s
17750K	68%	52.8M	0s
17800K	68%	127M	0s
17850K	68%	88.7M	0s
17900K	69%	47.5M	0s
17950K	69%	52.2M	0s
18000K	69%	82.8M	0s
18050K	69%	85.9M	0s
18100K	69%	152M	0s
18150K	70%	134M	0s
18200K	70%	272M	0s
18250K	70%	314M	0s
18300K	70%	304M	0s
18350K	70%	256M	0s
18400K	71%	245M	0s
18450K	71%	71.1M	0s
18500K	71%	79.6M	0s
18550K	71%	79.3M	0s
18600K	71%	89.2M	0s
18650K	71%	90.0M	0s
18700K	72%	119M	0s
18750K	72%	82.9M	0s
18800K	72%	130M	0s
18850K	72%	276M	0s
18900K	72%	314M	0s
18950K	73%	217M	0s
19000K	73%	298M	0s
19050K	73%	302M	0s
19100K	73%	274M	0s
19150K	73%	250M	0s
19200K	74%	302M	0s
19250K	74%	231M	0s
19300K	74%	282M	0s
19350K	74%	252M	0s
19400K	74%	269M	0s
19450K	75%	274M	0s
19500K	75%	305M	0s
19550K	75%	251M	0s
19600K	75%	290M	0s
19650K	75%	298M	0s
19700K	76%	314M	0s
19750K	76%	222M	0s
19800K	76%	228M	0s
19850K	76%	54.0M	0s
19900K	76%	67.7M	0s

19950K	76%	144M	0s
20000K	77%	173M	0s
20050K	77%	180M	0s
20100K	77%	180M	0s
20150K	77%	117M	0s
20200K	77%	158M	0s
20250K	78%	207M	0s
20300K	78%	305M	0s
20350K	78%	186M	0s
20400K	78%	124M	0s
20450K	78%	160M	0s
20500K	79%	204M	0s
20550K	79%	158M	0s
20600K	79%	206M	0s
20650K	79%	190M	0s
20700K	79%	208M	0s
20750K	80%	52.4M	0s
20800K	80%	97.9M	0s
20850K	80%	90.7M	0s
20900K	80%	122M	0s
20950K	80%	81.2M	0s
21000K	81%	303M	0s
21050K	81%	317M	0s
21100K	81%	183M	0s
21150K	81%	165M	0s
21200K	81%	193M	0s
21250K	82%	181M	0s
21300K	82%	163M	0s
21350K	82%	173M	0s
21400K	82%	192M	0s
21450K	82%	180M	0s
21500K	82%	166M	0s
21550K	83%	55.0M	0s
21600K	83%	138M	0s
21650K	83%	146M	0s
21700K	83%	309M	0s
21750K	83%	256M	0s
21800K	84%	313M	0s
21850K	84%	275M	0s
21900K	84%	282M	0s
21950K	84%	285M	0s
22000K	84%	309M	0s
22050K	85%	319M	0s
22100K	85%	304M	0s
22150K	85%	116M	0s
22200K	85%	169M	0s
22250K	85%	163M	0s
22300K	86%	46.1M	0s
22350K	86%	154M	0s
22400K	86%	170M	0s
22450K	86%	310M	0s

22500K	86%	309M	0s
22550K	87%	258M	0s
22600K	87%	137M	0s
22650K	87%	171M	0s
22700K	87%	200M	0s
22750K	87%	226M	0s
22800K	87%	138M	0s
22850K	88%	174M	0s
22900K	88%	66.8M	0s
22950K	88%	76.6M	0s
23000K	88%	104M	0s
23050K	88%	99.7M	0s
23100K	89%	294M	0s
23150K	89%	283M	0s
23200K	89%	315M	0s
23250K	89%	308M	0s
23300K	89%	308M	0s
23350K	90%	244M	0s
23400K	90%	267M	0s
23450K	90%	309M	0s
23500K	90%	153M	0s
23550K	90%	66.3M	0s
23600K	91%	182M	0s
23650K	91%	182M	0s
23700K	91%	172M	0s
23750K	91%	148M	0s
23800K	91%	118M	0s
23850K	92%	267M	0s
23900K	92%	315M	0s
23950K	92%	237M	0s
24000K	92%	117M	0s
24050K	92%	178M	0s
24100K	92%	175M	0s
24150K	93%	50.1M	0s
24200K	93%	91.4M	0s
24250K	93%	127M	0s
24300K	93%	81.4M	0s
24350K	93%	69.2M	0s
24400K	94%	154M	0s
24450K	94%	273M	0s
24500K	94%	298M	0s
24550K	94%	251M	0s
24600K	94%	307M	0s
24650K	95%	75.8M	0s
24700K	95%	298M	0s
24750K	95%	124M	0s
24800K	95%	302M	0s
24850K	95%	170M	0s
24900K	96%	306M	0s
24950K	96%	253M	0s
25000K	96%	304M	0s

25050K	96%	309M 0s
25100K	96%	306M 0s
25150K	97%	274M 0s
25200K	97%	251M 0s
25250K	97%	303M 0s
25300K	97%	309M 0s
25350K	97%	254M 0s
25400K	97%	297M 0s
25450K	98%	308M 0s
25500K	98%	272M 0s
25550K	98%	203M 0s
25600K	98%	53.2M 0s
25650K	98%	158M 0s
25700K	99%	157M 0s
25750K	99%	135M 0s
25800K	99%	166M 0s
25850K	99%	165M 0s
25900K	99%	158M 0s
25950K	100%	148M=0.2s

2020-05-31 14:09:51 (108 MB/s) - 'title.episode.tsv.gz' saved [26597530/26597530]

/bin/bash: line 1: fg: no job control

```
In [14]: title_episode = spark.read.option("sep", "\t").csv('file:/databricks/driver/ti
title_episode.tsv', header=True, inferSchema = True)
title_episode.cache()
title_episode.show(3)
```

+-----+-----+-----+-----+			
tconst parentTconst seasonNumber episodeNumber			
+-----+-----+-----+-----+			
tt0041951	tt0041038	1	9
tt0042816	tt0989125	1	17
tt0042889	tt0989125	\N	\N
+-----+-----+-----+-----+			
only showing top 3 rows			

```
In [15]: %sh wget https://datasets.imdbws.com/title.principals.tsv.gz  
%sh  
gunzip title.principals.tsv.gz
```

```
--2020-05-31 14:10:02-- https://datasets.imdbws.com/title.principals.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.26, 13.224.13.32,
13.224.13.37, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.26|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 323587283 (309M) [binary/octet-stream]
Saving to: 'title.principals.tsv.gz'
```

0K	0%	3.74M	83s
50K	0%	7.82M	61s
100K	0%	13.3M	48s
150K	0%	12.2M	43s
200K	0%	17.5M	38s
250K	0%	27.8M	33s
300K	0%	21.2M	30s
350K	0%	25.0M	28s
400K	0%	39.9M	26s
450K	0%	27.6M	24s
500K	0%	45.5M	23s
550K	0%	37.0M	22s
600K	0%	61.1M	20s
650K	0%	54.5M	19s
700K	0%	38.9M	19s
750K	0%	76.3M	18s
800K	0%	53.8M	17s
850K	0%	87.2M	16s
900K	0%	59.0M	16s
950K	0%	57.7M	15s
1000K	0%	123M	14s
1050K	0%	70.5M	14s
1100K	0%	98.4M	14s
1150K	0%	62.9M	13s
1200K	0%	95.5M	13s
1250K	0%	128M	12s
1300K	0%	114M	12s
1350K	0%	69.1M	12s
1400K	0%	137M	11s
1450K	0%	255M	11s
1500K	0%	73.2M	11s
1550K	0%	117M	11s
1600K	0%	104M	10s
1650K	0%	134M	10s
1700K	0%	141M	10s
1750K	0%	137M	10s
1800K	0%	112M	9s
1850K	0%	149M	9s
1900K	0%	129M	9s
1950K	0%	238M	9s
2000K	0%	106M	9s
2050K	0%	159M	9s

2100K	0%	146M	8s
2150K	0%	150M	8s
2200K	0%	149M	8s
2250K	0%	185M	8s
2300K	0%	121M	8s
2350K	0%	143M	8s
2400K	0%	174M	8s
2450K	0%	133M	8s
2500K	0%	142M	7s
2550K	0%	198M	7s
2600K	0%	202M	7s
2650K	0%	201M	7s
2700K	0%	162M	7s
2750K	0%	160M	7s
2800K	0%	112M	7s
2850K	0%	162M	7s
2900K	0%	125M	7s
2950K	0%	153M	7s
3000K	0%	161M	7s
3050K	0%	145M	6s
3100K	0%	123M	6s
3150K	1%	183M	6s
3200K	1%	198M	6s
3250K	1%	270M	6s
3300K	1%	165M	6s
3350K	1%	299M	6s
3400K	1%	219M	6s
3450K	1%	282M	6s
3500K	1%	154M	6s
3550K	1%	125M	6s
3600K	1%	148M	6s
3650K	1%	154M	6s
3700K	1%	140M	6s
3750K	1%	140M	6s
3800K	1%	193M	6s
3850K	1%	135M	5s
3900K	1%	129M	5s
3950K	1%	143M	5s
4000K	1%	159M	5s
4050K	1%	158M	5s
4100K	1%	157M	5s
4150K	1%	271M	5s
4200K	1%	206M	5s
4250K	1%	187M	5s
4300K	1%	172M	5s
4350K	1%	202M	5s
4400K	1%	271M	5s
4450K	1%	165M	5s
4500K	1%	204M	5s
4550K	1%	276M	5s
4600K	1%	192M	5s

4650K	1%	296M	5s
4700K	1%	84.4M	5s
4750K	1%	150M	5s
4800K	1%	149M	5s
4850K	1%	158M	5s
4900K	1%	139M	5s
4950K	1%	155M	5s
5000K	1%	154M	5s
5050K	1%	136M	5s
5100K	1%	125M	5s
5150K	1%	157M	5s
5200K	1%	221M	5s
5250K	1%	264M	4s
5300K	1%	78.8M	4s
5350K	1%	153M	4s
5400K	1%	155M	4s
5450K	1%	159M	4s
5500K	1%	130M	4s
5550K	1%	156M	4s
5600K	1%	184M	4s
5650K	1%	127M	4s
5700K	1%	132M	4s
5750K	1%	186M	4s
5800K	1%	134M	4s
5850K	1%	156M	4s
5900K	1%	123M	4s
5950K	1%	142M	4s
6000K	1%	151M	4s
6050K	1%	158M	4s
6100K	1%	137M	4s
6150K	1%	157M	4s
6200K	1%	158M	4s
6250K	1%	139M	4s
6300K	2%	123M	4s
6350K	2%	156M	4s
6400K	2%	158M	4s
6450K	2%	155M	4s
6500K	2%	181M	4s
6550K	2%	258M	4s
6600K	2%	230M	4s
6650K	2%	279M	4s
6700K	2%	229M	4s
6750K	2%	195M	4s
6800K	2%	263M	4s
6850K	2%	272M	4s
6900K	2%	183M	4s
6950K	2%	216M	4s
7000K	2%	261M	4s
7050K	2%	215M	4s
7100K	2%	123M	4s
7150K	2%	107M	4s

7200K	2%	125M	4s
7250K	2%	106M	4s
7300K	2%	137M	4s
7350K	2%	170M	4s
7400K	2%	198M	4s
7450K	2%	239M	4s
7500K	2%	218M	4s
7550K	2%	196M	4s
7600K	2%	256M	4s
7650K	2%	297M	4s
7700K	2%	183M	4s
7750K	2%	210M	4s
7800K	2%	96.3M	4s
7850K	2%	116M	4s
7900K	2%	130M	4s
7950K	2%	161M	4s
8000K	2%	145M	4s
8050K	2%	155M	4s
8100K	2%	146M	4s
8150K	2%	145M	4s
8200K	2%	159M	4s
8250K	2%	163M	4s
8300K	2%	144M	4s
8350K	2%	267M	4s
8400K	2%	178M	3s
8450K	2%	109M	3s
8500K	2%	138M	3s
8550K	2%	160M	3s
8600K	2%	151M	3s
8650K	2%	159M	3s
8700K	2%	132M	3s
8750K	2%	178M	3s
8800K	2%	138M	3s
8850K	2%	154M	3s
8900K	2%	158M	3s
8950K	2%	144M	3s
9000K	2%	143M	3s
9050K	2%	151M	3s
9100K	2%	101M	3s
9150K	2%	184M	3s
9200K	2%	259M	3s
9250K	2%	237M	3s
9300K	2%	181M	3s
9350K	2%	258M	3s
9400K	2%	246M	3s
9450K	3%	225M	3s
9500K	3%	188M	3s
9550K	3%	258M	3s
9600K	3%	310M	3s
9650K	3%	193M	3s
9700K	3%	208M	3s

9750K	3%	236M	3s
9800K	3%	92.8M	3s
9850K	3%	134M	3s
9900K	3%	138M	3s
9950K	3%	143M	3s
10000K	3%	148M	3s
10050K	3%	149M	3s
10100K	3%	140M	3s
10150K	3%	154M	3s
10200K	3%	156M	3s
10250K	3%	146M	3s
10300K	3%	176M	3s
10350K	3%	205M	3s
10400K	3%	113M	3s
10450K	3%	148M	3s
10500K	3%	126M	3s
10550K	3%	153M	3s
10600K	3%	147M	3s
10650K	3%	150M	3s
10700K	3%	113M	3s
10750K	3%	159M	3s
10800K	3%	148M	3s
10850K	3%	157M	3s
10900K	3%	143M	3s
10950K	3%	147M	3s
11000K	3%	148M	3s
11050K	3%	154M	3s
11100K	3%	130M	3s
11150K	3%	159M	3s
11200K	3%	149M	3s
11250K	3%	147M	3s
11300K	3%	146M	3s
11350K	3%	141M	3s
11400K	3%	149M	3s
11450K	3%	153M	3s
11500K	3%	124M	3s
11550K	3%	133M	3s
11600K	3%	160M	3s
11650K	3%	154M	3s
11700K	3%	125M	3s
11750K	3%	147M	3s
11800K	3%	153M	3s
11850K	3%	152M	3s
11900K	3%	127M	3s
11950K	3%	161M	3s
12000K	3%	155M	3s
12050K	3%	140M	3s
12100K	3%	146M	3s
12150K	3%	138M	3s
12200K	3%	154M	3s
12250K	3%	148M	3s

12300K	3%	137M	3s
12350K	3%	134M	3s
12400K	3%	160M	3s
12450K	3%	148M	3s
12500K	3%	146M	3s
12550K	3%	152M	3s
12600K	4%	157M	3s
12650K	4%	147M	3s
12700K	4%	128M	3s
12750K	4%	146M	3s
12800K	4%	182M	3s
12850K	4%	137M	3s
12900K	4%	140M	3s
12950K	4%	154M	3s
13000K	4%	139M	3s
13050K	4%	136M	3s
13100K	4%	127M	3s
13150K	4%	154M	3s
13200K	4%	148M	3s
13250K	4%	160M	3s
13300K	4%	146M	3s
13350K	4%	142M	3s
13400K	4%	194M	3s
13450K	4%	247M	3s
13500K	4%	163M	3s
13550K	4%	197M	3s
13600K	4%	88.6M	3s
13650K	4%	160M	3s
13700K	4%	135M	3s
13750K	4%	141M	3s
13800K	4%	140M	3s
13850K	4%	149M	3s
13900K	4%	135M	3s
13950K	4%	149M	3s
14000K	4%	154M	3s
14050K	4%	153M	3s
14100K	4%	143M	3s
14150K	4%	144M	3s
14200K	4%	162M	3s
14250K	4%	147M	3s
14300K	4%	130M	3s
14350K	4%	148M	3s
14400K	4%	152M	3s
14450K	4%	145M	3s
14500K	4%	132M	3s
14550K	4%	126M	3s
14600K	4%	156M	3s
14650K	4%	152M	3s
14700K	4%	123M	3s
14750K	4%	191M	3s
14800K	4%	135M	3s

14850K	4%	153M	3s
14900K	4%	128M	3s
14950K	4%	159M	3s
15000K	4%	169M	3s
15050K	4%	140M	3s
15100K	4%	174M	3s
15150K	4%	155M	3s
15200K	4%	147M	3s
15250K	4%	150M	3s
15300K	4%	142M	3s
15350K	4%	159M	3s
15400K	4%	141M	3s
15450K	4%	144M	3s
15500K	4%	131M	3s
15550K	4%	168M	3s
15600K	4%	233M	3s
15650K	4%	189M	3s
15700K	4%	251M	3s
15750K	4%	239M	3s
15800K	5%	124M	3s
15850K	5%	261M	3s
15900K	5%	179M	3s
15950K	5%	295M	3s
16000K	5%	187M	3s
16050K	5%	185M	3s
16100K	5%	180M	3s
16150K	5%	247M	3s

*** WARNING: skipped 430920 bytes of output ***

299700K	94%	141M	0s
299750K	94%	95.1M	0s
299800K	94%	134M	0s
299850K	94%	171M	0s
299900K	94%	185M	0s
299950K	94%	133M	0s
300000K	94%	98.0M	0s
300050K	94%	110M	0s
300100K	94%	109M	0s
300150K	94%	119M	0s
300200K	95%	139M	0s
300250K	95%	172M	0s
300300K	95%	187M	0s
300350K	95%	192M	0s
300400K	95%	191M	0s
300450K	95%	153M	0s
300500K	95%	111M	0s
300550K	95%	115M	0s
300600K	95%	98.6M	0s
300650K	95%	121M	0s
300700K	95%	99.0M	0s

300750K	95%	132M	0s
300800K	95%	130M	0s
300850K	95%	91.0M	0s
300900K	95%	125M	0s
300950K	95%	95.6M	0s
301000K	95%	128M	0s
301050K	95%	121M	0s
301100K	95%	98.4M	0s
301150K	95%	113M	0s
301200K	95%	86.6M	0s
301250K	95%	114M	0s
301300K	95%	94.2M	0s
301350K	95%	178M	0s
301400K	95%	189M	0s
301450K	95%	172M	0s
301500K	95%	168M	0s
301550K	95%	186M	0s
301600K	95%	190M	0s
301650K	95%	88.4M	0s
301700K	95%	126M	0s
301750K	95%	87.6M	0s
301800K	95%	94.7M	0s
301850K	95%	114M	0s
301900K	95%	96.2M	0s
301950K	95%	108M	0s
302000K	95%	130M	0s
302050K	95%	79.3M	0s
302100K	95%	90.3M	0s
302150K	95%	202M	0s
302200K	95%	69.2M	0s
302250K	95%	89.9M	0s
302300K	95%	94.0M	0s
302350K	95%	170M	0s
302400K	95%	179M	0s
302450K	95%	156M	0s
302500K	95%	189M	0s
302550K	95%	183M	0s
302600K	95%	183M	0s
302650K	95%	170M	0s
302700K	95%	95.9M	0s
302750K	95%	130M	0s
302800K	95%	92.2M	0s
302850K	95%	86.8M	0s
302900K	95%	124M	0s
302950K	95%	118M	0s
303000K	95%	130M	0s
303050K	95%	167M	0s
303100K	95%	188M	0s
303150K	95%	193M	0s
303200K	95%	126M	0s
303250K	95%	141M	0s

303300K	95%	118M	0s
303350K	96%	184M	0s
303400K	96%	127M	0s
303450K	96%	167M	0s
303500K	96%	184M	0s
303550K	96%	177M	0s
303600K	96%	125M	0s
303650K	96%	152M	0s
303700K	96%	193M	0s
303750K	96%	188M	0s
303800K	96%	186M	0s
303850K	96%	168M	0s
303900K	96%	194M	0s
303950K	96%	186M	0s
304000K	96%	182M	0s
304050K	96%	76.8M	0s
304100K	96%	128M	0s
304150K	96%	90.0M	0s
304200K	96%	129M	0s
304250K	96%	101M	0s
304300K	96%	125M	0s
304350K	96%	120M	0s
304400K	96%	189M	0s
304450K	96%	153M	0s
304500K	96%	190M	0s
304550K	96%	100M	0s
304600K	96%	93.6M	0s
304650K	96%	90.8M	0s
304700K	96%	125M	0s
304750K	96%	192M	0s
304800K	96%	191M	0s
304850K	96%	88.9M	0s
304900K	96%	91.5M	0s
304950K	96%	128M	0s
305000K	96%	189M	0s
305050K	96%	173M	0s
305100K	96%	194M	0s
305150K	96%	196M	0s
305200K	96%	78.4M	0s
305250K	96%	104M	0s
305300K	96%	92.0M	0s
305350K	96%	94.8M	0s
305400K	96%	133M	0s
305450K	96%	82.2M	0s
305500K	96%	189M	0s
305550K	96%	191M	0s
305600K	96%	189M	0s
305650K	96%	86.3M	0s
305700K	96%	119M	0s
305750K	96%	95.4M	0s
305800K	96%	121M	0s

305850K	96%	170M	0s
305900K	96%	179M	0s
305950K	96%	188M	0s
306000K	96%	94.3M	0s
306050K	96%	92.3M	0s
306100K	96%	187M	0s
306150K	96%	188M	0s
306200K	96%	181M	0s
306250K	96%	192M	0s
306300K	96%	203M	0s
306350K	96%	105M	0s
306400K	96%	137M	0s
306450K	96%	82.9M	0s
306500K	97%	134M	0s
306550K	97%	216M	0s
306600K	97%	217M	0s
306650K	97%	105M	0s
306700K	97%	204M	0s
306750K	97%	210M	0s
306800K	97%	189M	0s
306850K	97%	169M	0s
306900K	97%	196M	0s
306950K	97%	196M	0s
307000K	97%	75.3M	0s
307050K	97%	109M	0s
307100K	97%	119M	0s
307150K	97%	183M	0s
307200K	97%	190M	0s
307250K	97%	157M	0s
307300K	97%	129M	0s
307350K	97%	88.0M	0s
307400K	97%	132M	0s
307450K	97%	170M	0s
307500K	97%	184M	0s
307550K	97%	187M	0s
307600K	97%	190M	0s
307650K	97%	96.8M	0s
307700K	97%	112M	0s
307750K	97%	103M	0s
307800K	97%	119M	0s
307850K	97%	90.6M	0s
307900K	97%	113M	0s
307950K	97%	91.7M	0s
308000K	97%	184M	0s
308050K	97%	155M	0s
308100K	97%	196M	0s
308150K	97%	93.9M	0s
308200K	97%	125M	0s
308250K	97%	88.6M	0s
308300K	97%	130M	0s
308350K	97%	121M	0s

308400K	97%	173M	0s
308450K	97%	92.2M	0s
308500K	97%	134M	0s
308550K	97%	103M	0s
308600K	97%	111M	0s
308650K	97%	170M	0s
308700K	97%	187M	0s
308750K	97%	164M	0s
308800K	97%	105M	0s
308850K	97%	155M	0s
308900K	97%	194M	0s
308950K	97%	178M	0s
309000K	97%	189M	0s
309050K	97%	99.9M	0s
309100K	97%	109M	0s
309150K	97%	103M	0s
309200K	97%	128M	0s
309250K	97%	110M	0s
309300K	97%	188M	0s
309350K	97%	84.9M	0s
309400K	97%	132M	0s
309450K	97%	106M	0s
309500K	97%	190M	0s
309550K	97%	187M	0s
309600K	97%	191M	0s
309650K	98%	154M	0s
309700K	98%	101M	0s
309750K	98%	112M	0s
309800K	98%	117M	0s
309850K	98%	166M	0s
309900K	98%	189M	0s
309950K	98%	187M	0s
310000K	98%	188M	0s
310050K	98%	67.3M	0s
310100K	98%	188M	0s
310150K	98%	187M	0s
310200K	98%	185M	0s
310250K	98%	173M	0s
310300K	98%	76.2M	0s
310350K	98%	112M	0s
310400K	98%	82.7M	0s
310450K	98%	86.8M	0s
310500K	98%	92.1M	0s
310550K	98%	187M	0s
310600K	98%	185M	0s
310650K	98%	116M	0s
310700K	98%	182M	0s
310750K	98%	189M	0s
310800K	98%	189M	0s
310850K	98%	155M	0s
310900K	98%	73.0M	0s

310950K	98%	130M	0s
311000K	98%	86.8M	0s
311050K	98%	163M	0s
311100K	98%	189M	0s
311150K	98%	188M	0s
311200K	98%	101M	0s
311250K	98%	90.3M	0s
311300K	98%	178M	0s
311350K	98%	187M	0s
311400K	98%	190M	0s
311450K	98%	72.8M	0s
311500K	98%	126M	0s
311550K	98%	98.1M	0s
311600K	98%	135M	0s
311650K	98%	149M	0s
311700K	98%	186M	0s
311750K	98%	183M	0s
311800K	98%	186M	0s
311850K	98%	168M	0s
311900K	98%	101M	0s
311950K	98%	181M	0s
312000K	98%	185M	0s
312050K	98%	157M	0s
312100K	98%	135M	0s
312150K	98%	94.1M	0s
312200K	98%	131M	0s
312250K	98%	87.0M	0s
312300K	98%	185M	0s
312350K	98%	189M	0s
312400K	98%	191M	0s
312450K	98%	76.6M	0s
312500K	98%	129M	0s
312550K	98%	190M	0s
312600K	98%	204M	0s
312650K	98%	171M	0s
312700K	98%	25.6M	0s
312750K	98%	188M	0s
312800K	99%	69.9M	0s
312850K	99%	155M	0s
312900K	99%	95.9M	0s
312950K	99%	106M	0s
313000K	99%	85.9M	0s
313050K	99%	64.7M	0s
313100K	99%	74.5M	0s
313150K	99%	81.2M	0s
313200K	99%	153M	0s
313250K	99%	89.3M	0s
313300K	99%	82.9M	0s
313350K	99%	142M	0s
313400K	99%	188M	0s
313450K	99%	163M	0s

313500K	99%	77.4M	0s
313550K	99%	155M	0s
313600K	99%	190M	0s
313650K	99%	145M	0s
313700K	99%	172M	0s
313750K	99%	194M	0s
313800K	99%	187M	0s
313850K	99%	166M	0s
313900K	99%	189M	0s
313950K	99%	154M	0s
314000K	99%	88.3M	0s
314050K	99%	110M	0s
314100K	99%	194M	0s
314150K	99%	188M	0s
314200K	99%	115M	0s
314250K	99%	172M	0s
314300K	99%	196M	0s
314350K	99%	184M	0s
314400K	99%	191M	0s
314450K	99%	156M	0s
314500K	99%	190M	0s
314550K	99%	199M	0s
314600K	99%	86.6M	0s
314650K	99%	169M	0s
314700K	99%	191M	0s
314750K	99%	194M	0s
314800K	99%	188M	0s
314850K	99%	10.4M	0s
314900K	99%	167M	0s
314950K	99%	174M	0s
315000K	99%	175M	0s
315050K	99%	152M	0s
315100K	99%	169M	0s
315150K	99%	171M	0s
315200K	99%	172M	0s
315250K	99%	151M	0s
315300K	99%	184M	0s
315350K	99%	189M	0s
315400K	99%	185M	0s
315450K	99%	166M	0s
315500K	99%	196M	0s
315550K	99%	189M	0s
315600K	99%	186M	0s
315650K	99%	131M	0s
315700K	99%	183M	0s
315750K	99%	193M	0s
315800K	99%	189M	0s
315850K	99%	171M	0s
315900K	99%	173M	0s
315950K	99%	189M	0s
316000K	...	100%	53.9M=2.4s	

2020-05-31 14:10:04 (129 MB/s) - 'title.principals.tsv.gz' saved [323587283/323587283]

/bin/bash: line 1: fg: no job control

```
In [16]: title_principals = spark.read.option("sep", "\t").csv('file:/databricks/drive
r/title.principals.tsv', header=True, inferSchema = True)
title_principals.cache()
title_principals.show(3)
```

```
+-----+-----+-----+-----+-----+-----+
  tconst|ordering|  nconst|      category|      job|characters|
+-----+-----+-----+-----+-----+-----+
tt0000001|      1|nm1588970|      self|      \N| ["Self"]|
tt0000001|      2|nm0005690|    director|      \N|      \N|
tt0000001|      3|nm0374658|cinematographer|director of photo...|      \N|
+-----+-----+-----+-----+-----+-----+
only showing top 3 rows
```

```
In [17]: %sh wget https://datasets.imdbws.com/title.ratings.tsv.gz  
%sh  
gunzip title.ratings.tsv.gz
```

```
--2020-05-31 14:11:40-- https://datasets.imdbws.com/title.ratings.tsv.gz
Resolving datasets.imdbws.com (datasets.imdbws.com)... 13.224.13.32, 13.224.13.37,
13.224.13.54, ...
Connecting to datasets.imdbws.com (datasets.imdbws.com)|13.224.13.32|:443... connec
ted.
HTTP request sent, awaiting response... 200 OK
Length: 5169313 (4.9M) [binary/octet-stream]
Saving to: 'title.ratings.tsv.gz'
```

0K	0%	4.34M	1s
50K	1%	6.17M	1s
100K	2%	13.2M	1s
150K	3%	11.8M	1s
200K	4%	17.9M	1s
250K	5%	28.9M	0s
300K	6%	20.9M	0s
350K	7%	25.1M	0s
400K	8%	38.7M	0s
450K	9%	37.5M	0s
500K	10%	33.5M	0s
550K	11%	46.6M	0s
600K	12%	33.0M	0s
650K	13%	59.6M	0s
700K	14%	45.2M	0s
750K	15%	17.4M	0s
800K	16%	162M	0s
850K	17%	128M	0s
900K	18%	181M	0s
950K	19%	133M	0s
1000K	20%	121M	0s
1050K	21%	144M	0s
1100K	22%	89.4M	0s
1150K	23%	90.2M	0s
1200K	24%	65.8M	0s
1250K	25%	154M	0s
1300K	26%	108M	0s
1350K	27%	72.0M	0s
1400K	28%	98.8M	0s
1450K	29%	110M	0s
1500K	30%	145M	0s
1550K	31%	20.5M	0s
1600K	32%	171M	0s
1650K	33%	133M	0s
1700K	34%	149M	0s
1750K	35%	147M	0s
1800K	36%	119M	0s
1850K	37%	194M	0s
1900K	38%	202M	0s
1950K	39%	205M	0s
2000K	40%	144M	0s
2050K	41%	161M	0s

2100K	42%	189M	0s
2150K	43%	165M	0s
2200K	44%	141M	0s
2250K	45%	184M	0s
2300K	46%	188M	0s
2350K	47%	193M	0s
2400K	48%	155M	0s
2450K	49%	13.2M	0s
2500K	50%	124M	0s
2550K	51%	147M	0s
2600K	52%	150M	0s
2650K	53%	133M	0s
2700K	54%	136M	0s
2750K	55%	152M	0s
2800K	56%	128M	0s
2850K	57%	153M	0s
2900K	58%	155M	0s
2950K	59%	147M	0s
3000K	60%	117M	0s
3050K	61%	143M	0s
3100K	62%	139M	0s
3150K	63%	139M	0s
3200K	64%	136M	0s
3250K	65%	152M	0s
3300K	66%	233M	0s
3350K	67%	189M	0s
3400K	68%	166M	0s
3450K	69%	205M	0s
3500K	70%	210M	0s
3550K	71%	184M	0s
3600K	72%	155M	0s
3650K	73%	294M	0s
3700K	74%	177M	0s
3750K	75%	248M	0s
3800K	76%	240M	0s
3850K	77%	238M	0s
3900K	78%	290M	0s
3950K	79%	299M	0s
4000K	80%	268M	0s
4050K	81%	208M	0s
4100K	82%	200M	0s
4150K	83%	212M	0s
4200K	84%	182M	0s
4250K	85%	296M	0s
4300K	86%	184M	0s
4350K	87%	203M	0s
4400K	88%	265M	0s
4450K	89%	184M	0s
4500K	90%	193M	0s
4550K	91%	284M	0s
4600K	92%	170M	0s

```

4650K ..... 93% 290M 0s
4700K ..... 94% 294M 0s
4750K ..... 95% 299M 0s
4800K ..... 96% 66.5M 0s
4850K ..... 97% 155M 0s
4900K ..... 98% 143M 0s
4950K ..... 99% 130M 0s
5000K ..... 100% 148M=0.08s

```

2020-05-31 14:11:40 (62.3 MB/s) - 'title.ratings.tsv.gz' saved [5169313/5169313]

/bin/bash: line 1: fg: no job control

```

In [18]: title_ratings = spark.read.option("sep", "\t").csv('file:/databricks/driver/ti
tle.ratings.tsv', header=True, inferSchema = True)
title_ratings.cache()
title_ratings.show(3)

```

```

+-----+-----+-----+
  tconst|averageRating|numVotes|
+-----+-----+-----+
tt0000001|      5.6|    1619|
tt0000002|      6.0|     198|
tt0000003|      6.5|    1301|
+-----+-----+-----+
only showing top 3 rows

```

Network Inference, Let's build a network

In the following questions you will look to summarise the data and build a network. We want to examine a network that abstracts how actors and actress are related through their co-participation in movies. To that end perform the following steps:

Q1 Create a DataFrame that combines the information on each of the titles (i.e., movies, tv-shows, etc ...) and the information on the participants in those movies (i.e., actors, directors, etc ...), make sure the actual names of the movies and participants are included. It may be worth reviewing the following questions to see how this dataframe will be used.

How many rows does your dataframe have?

```

In [20]: # filtered = title_principals.join(title_basics,['tconst'],how='left_outer').j
oin(names_basics,['nconst'],how='left_outer')
tconts_join = title_principals.join(title_basics, title_principals.tconst == t
itle_basics.tconst,how='left').drop(title_basics.tconst)
filtered = tconts_join.join(names_basics, tconts_join.nconst == names_basics.n
const,how='left').drop(names_basics.nconst)

```

```
In [21]: print("This Dataframe has " + str(filtered.count()) + " rows")
```

This Dataframe has 39502500 rows

Q2 Create a new DataFrame based on the previous step, with the following removed:

1. Any participant that is not an actor or actress (as measured by the category column);
2. All adult movies;
3. All dead actors or actresses;
4. All actors or actresses born before 1920 or with no date of birth listed;
5. All titles that are not of the type movie.

How many rows does your dataframe have?

```
In [23]: df_filter = filtered.filter(filtered.category.isin(['actor', 'actress'])) \
        .filter(filtered.isAdult != 1) \
        .filter(filtered.deathYear == "\\N") \
        .filter(filtered.birthYear >= 1920) \
        .filter(filtered.birthYear.isNotNull()) \
        .filter(filtered.titleType == 'movie')
```

```
In [24]: print("This Dataframe has " + str(df_filter.count()) + " rows")
```

This Dataframe has 451106 rows

Q3 Convert the above DataFrame to an RDD (you can use `.rdd` to convert a dataframe to an RDD of row objects). Use map and reduce to create a paired RDD which counts how many movies each actor / actress appears in.

Display names of the top 10 actors/actresses according to the number of movies in which they appeared. Be careful to deal with different actors / actresses with the same name, these could be different people.

In [26]: `df_filter.columns`

```
Out[21]: ['tconst',
          'ordering',
          'nconst',
          'category',
          'job',
          'characters',
          'titleType',
          'primaryTitle',
          'originalTitle',
          'isAdult',
          'startYear',
          'endYear',
          'runtimeMinutes',
          'genres',
          'primaryName',
          'birthYear',
          'deathYear',
          'primaryProfession',
          'knownForTitles']
```

In [27]: `Ids_df = df_filter.rdd.map(lambda x : ((x['nconst'], x['primaryName']), 1)) # map by ids "nconst" and "name"`
`df = Ids_df.reduceByKey(lambda x,y : x+y)`
`df_final = df.sortBy(lambda x: x[1], False)`

In [28]: `print("The top 10 actors/actresses according to the number of movies in which they appeared in are: ")`
`df_final.take(10)`

The top 10 actors/actresses according to the number of movies in which they appeared in are:

```
Out[23]: [('nm0103977', 'Brahmanandam'), 809),
          ('nm0007123', 'Mammootty'), 379),
          ('nm0482320', 'Mohanlal'), 343),
          ('nm0149822', 'Mithun Chakraborty'), 332),
          ('nm0007106', 'Shakti Kapoor'), 309),
          ('nm0415549', 'Jagathi Sreekumar'), 303),
          ('nm0035067', 'Cüneyt Arkin'), 294),
          ('nm0374974', 'Helen'), 281),
          ('nm0534867', 'Madhu'), 277),
          ('nm0004429', 'Dharmendra'), 270)]
```

Q4 Start with the dataframe from **Q2**. Generate a DataFrame that lists all links of your network. Here we shall consider that a link connects a pair of actors/actresses if they participated in at least one movie together (actors / actresses should be represented by their unique ID's). For every link we then need anytime a pair of actors were together in a movie as a link in each direction (A -> B and B -> A). However links should be distinct we do not need duplicates when two actors worked together in several movies.

```
In [30]: actors_rdd_1 = df_filter.rdd.map(list).map(lambda x : (x[0], x[2])) #tconst and
nconst
actors_rdd_2 = df_filter.rdd.map(list).map(lambda x : (x[0], x[2]))
actors_rdd_1.take(5)
```

```
Out[24]: [('tt0110116', 'nm0000198'),
('tt1345836', 'nm0000198'),
('tt0097125', 'nm0000198'),
('tt3239932', 'nm0000198'),
('tt0208874', 'nm0000198')]
```

```
In [31]: actors_rdd_1.join(actors_rdd_2).take(10)
```

```
Out[25]: [('tt2104129', ('nm0048122', 'nm0048122')),
('tt2104129', ('nm0048122', 'nm0221697')),
('tt2104129', ('nm0221697', 'nm0048122')),
('tt2104129', ('nm0221697', 'nm0221697')),
('tt0226612', ('nm0549603', 'nm0549603')),
('tt1555367', ('nm0998254', 'nm0998254')),
('tt0096129', ('nm0527543', 'nm0527543')),
('tt0096129', ('nm0527543', 'nm0295429')),
('tt0096129', ('nm0527543', 'nm0087787')),
('tt0096129', ('nm0295429', 'nm0527543'))]
```

```
In [32]: from pyspark.sql import Row
dist_pairs = actors_rdd_1.join(actors_rdd_2).map(lambda x : Row(x[1][0], x[1][
1])).filter(lambda x: x[0] != x[1]).distinct()
```

```
In [33]: dist_pairs.take(1)
```

```
Out[27]: [<Row(nm3216408, nm0453304)>]
```

Q5 Compute the page rank of each actor. This can be done using GraphFrames or by using RDDs and the iterative implementation of the PageRank algorithm. Do not take more than 5 iterations and use reset probability = 0.1.

List the top 10 actors / actresses by pagerank.

```
In [35]: from graphframes import *
from pyspark.sql.types import *
from pyspark.sql.types import StructField
```



```
In [36]: ActorA = StructField("ActorA",StringType(),True)
ActorB = StructField("ActorB",StringType(),True)

df_actors_link = sqlContext.createDataFrame(dist_pairs, StructType([ActorA, ActorB])).persist()
df_actors_link.show(1)
```

```
+-----+-----+
ActorA| ActorB|
+-----+-----+
nm3216408|nm0453304|
+-----+-----+
only showing top 1 row
```

```
In [37]: df_cols = df_actors_link.select(df_actors_link['ActorA']).selectExpr("ActorA as id").distinct()
Rank_link = df_actors_link.selectExpr("ActorA as src", "ActorB as dst")
```

```
In [38]: import graphframes.graphframe as gfm  
  
ourGraph = gfm.GraphFrame(df_cols, Rank_link)  
ourGraph.vertices.show()  
ourGraph.edges.show()
```

```
+-----+
```

```
id|
```

```
+-----+
```

```
nm0246570|
```

```
nm0352203|
```

```
nm0005258|
```

```
nm0000767|
```

```
nm0000354|
```

```
nm0000198|
```

```
nm0225216|
```

```
nm0350453|
```

```
nm0380489|
```

```
nm0397721|
```

```
nm1563338|
```

```
nm0046770|
```

```
nm0696810|
```

```
nm0505949|
```

```
nm2612218|
```

```
nm2267821|
```

```
nm1415658|
```

```
nm0048122|
```

```
nm0005535|
```

```
nm0118840|
```

```
+-----+
```

```
only showing top 20 rows
```

```
+-----+-----+
```

```
src|dst|
```

```
+-----+-----+
```

```
nm3216408|nm0453304|
```

```
nm0544425|nm0000778|
```

```
nm2507102|nm0102403|
```

```
nm0668271|nm0001151|
```

```
nm0429385|nm0005541|
```

```
nm0059847|nm0036924|
```

```
nm0000665|nm0000546|
```

```
nm0879203|nm0863831|
```

```
nm1231899|nm0695177|
```

```
nm0666140|nm0744037|
```

```
nm0373571|nm0003620|
```

```
nm0341176|nm1260957|
```

```
nm0467601|nm1204831|
```

```
nm0077720|nm0139716|
```

```
nm0994324|nm0945475|
```

```
nm0142972|nm0530365|
```

```
nm0878714|nm0000437|
```

```
nm0004418|nm0883305|
```

```
nm0049190|nm0820544|
```

```
nm0001092|nm0001295|
```

```
+-----+-----+
```

only showing top 20 rows

```
In [39]: page_rank = ourGraph.pageRank(resetProbability=0.1, maxIter = 3)
page_rank.vertices.sort("pagerank", ascending = False).show(10)
#sortpagerank ascending = False).show(10) to get top 10
```

```
+-----+-----+
      id|      pagerank|
+-----+-----+
nm0000616|39.783048844356635|
nm0000514|23.417910767557924|
nm0001744|22.674253088847944|
nm0001803|20.265762987819404|
nm0004193| 17.40344156284936|
nm000448|17.401842412643557|
nm0001698| 16.98356025044102|
nm0261724|16.024450112153453|
nm0920460|15.662074823121074|
nm0001424| 15.52518575030298|
+-----+-----+
only showing top 10 rows
```

Q6: Create an RDD with the number of outDegrees for each actor. Display the top 10 by outDegrees.

```
In [41]: ourGraph.outDegrees.sort('outDegree', ascending = False).show(10)
```

```
+-----+-----+
      id|outDegree|
+-----+-----+
nm0000616|    438|
nm0000514|    289|
nm0000367|    263|
nm0001744|    261|
nm0945189|    253|
nm0451600|    239|
nm0149822|    232|
nm0001803|    231|
nm0874676|    227|
nm0938893|    225|
+-----+-----+
only showing top 10 rows
```

Let's play Kevin's own game

Q7 Start with the graphframe / dataframe you developed in the previous section. Using Spark GraphFrame and/or Spark Core library perform the following steps:

1. Identify the id of Kevin Bacon, there are two actors named 'Kevin Bacon', we will use the one with the highest degree, that is, the one that participated in most titles;
2. Estimate the shortest path between every actor/actress in the database and Kevin Bacon, keep a dataframe with a column that includes the number of steps to Kevin Bacon as you will need it later (this will require a little processing to get from the graphframes output);
3. Summarise the data, that is, count the number of actors at each number of degrees from Kevin Bacon (you will need to deal with actors unconnected to Kevin Bacon, if not connected to Kevin Bacon given these actors / actresses a score of 20). You could use the display() barchart functionality of databricks to easily display the distribution of the data.

Note: The solution time on this step can be ~15 minutes

```
In [43]: df_filter.columns
```

```
Out[34]: ['tconst',  
         'ordering',  
         'nconst',  
         'category',  
         'job',  
         'characters',  
         'titleType',  
         'primaryTitle',  
         'originalTitle',  
         'isAdult',  
         'startYear',  
         'endYear',  
         'runtimeMinutes',  
         'genres',  
         'primaryName',  
         'birthYear',  
         'deathYear',  
         'primaryProfession',  
         'knownForTitles']
```

```
In [44]: kb = df_filter.select("primaryName", "nconst")
```

```
In [45]: #1
distances = kb.filter(filtered.primaryName == "Kevin Bacon").groupBy('nconst')
            .count().sort('count', ascending = False)
distances.show(1)
```

```
+-----+-----+
|nconst|count|
+-----+-----+
|nm0000102| 49|
+-----+-----+
```

```
In [46]: shortest_path = ourGraph.shortestPaths(landmarks=["nm0000102"])
```

```
In [47]: from pyspark.sql.functions import explode_outer

distances_value = shortest_path.select("id", explode_outer("distances"))
distances_value.show()
```

```
+-----+-----+-----+
|id|key|value|
+-----+-----+-----+
|nm0057741|nm0000102| 4|
|nm0068551|nm0000102| 3|
|nm0309307|nm0000102| 3|
|nm2341870|nm0000102| 4|
|nm0067745|nm0000102| 4|
|nm1846114|nm0000102| 4|
|nm0190703|nm0000102| 4|
|nm0212603| null| null|
|nm0269419|nm0000102| 3|
|nm3773942|nm0000102| 4|
|nm0111600|nm0000102| 3|
|nm7007994|nm0000102| 4|
|nm2058568|nm0000102| 4|
|nm0319244|nm0000102| 3|
|nm0505971|nm0000102| 2|
|nm3836963|nm0000102| 5|
|nm0588022|nm0000102| 4|
|nm8510546|nm0000102| 5|
|nm0001796|nm0000102| 3|
|nm10364103|nm0000102| 5|
+-----+-----+-----+
only showing top 20 rows
```

```
In [48]: actors_per_distance = distances_value.groupBy('value').count()
         actors_per_distance.show()
```

```
+-----+-----+
value|count|
+-----+-----+
null| 6489|
  1|  126|
  6| 2265|
  3|18311|
  5|13109|
  9|   16|
  4|29059|
  8|   51|
  7|  328|
 10|   12|
 11|    4|
  2| 3302|
  0|    1|
+-----+-----+
```

Exploring the data with RDD's

Using RDDs and (not dataframes) answer the following questions (if you loaded your data into spark in a dataframe you can convert to an RDD of rows easily using `.rdd`) :

Hint: paired RDD's will be useful.

Q8 Movies can have multiple genres. Considering only titles of the type 'movie' what is the combination of genres that is the most popular (as measured by number of reviews)?

```
In [50]: movies_type = df_filter.join(title_ratings,['tconst'])
```

In [51]: `movies_type.columns`

```
Out[41]: ['tconst',
          'ordering',
          'nconst',
          'category',
          'job',
          'characters',
          'titleType',
          'primaryTitle',
          'originalTitle',
          'isAdult',
          'startYear',
          'endYear',
          'runtimeMinutes',
          'genres',
          'primaryName',
          'birthYear',
          'deathYear',
          'primaryProfession',
          'knownForTitles',
          'averageRating',
          'numVotes']
```

In [52]: `q8 = movies_type.select("genres", "numVotes")`
`title = q8.rdd.map(list).map(lambda x : (x[0], x[1]))`

In [53]: `final_title_result = title.reduceByKey(lambda x,y: x+y).sortBy(lambda x: x[1], False)`
`final_title_result.take(1)`

```
Out[43]: [('Action,Adventure,Sci-Fi', 164525767)]
```

Q9 Movies can have multiple genres. Considering only titles of the type 'movie', and movies with more than 500 ratings, what is the combination of genres that has the highest **average movie rating** (you can average the movie rating for each movie in that genre combination).

In [55]: `Rdd_q9 = title_basics.rdd.map(lambda x: (x[0], (x[1], x[-1])))`
`.join(title_ratings.rdd.map(lambda x: (x[0], (x[1], x[2]))))`
`Rdd_q9.take(5)`

```
Out[44]: [('tt0000001', (('short', 'Documentary,Short'), (5.6, 1619))),
          ('tt0000013', (('short', 'Documentary,Short'), (5.7, 1541))),
          ('tt0000031', (('short', 'Documentary,Short'), (5.5, 813))),
          ('tt0000038', (('short', 'Documentary,Short,Sport'), (4.1, 147))),
          ('tt0000057', (('short', 'Documentary,Short'), (4.7, 16)))]
```



```
In [56]: movie = Rdd_q9.map(lambda x: (x[1][0][0],x[1][0][1],x[1][1][0],x[1][1][1]))
movie.take(5)
```

```
Out[45]: [('short', 'Documentary,Short', 5.6, 1619),
('short', 'Documentary,Short', 5.7, 1541),
('short', 'Documentary,Short', 5.5, 813),
('short', 'Documentary,Short,Sport', 4.1, 147),
('short', 'Documentary,Short', 4.7, 16)]
```

```
In [57]: q9 = movie.filter(lambda x: (x[0] == 'movie') & (x[3] >= 500)).map(lambda x: (
x[1],x[2]))
q9.take(5)
```

```
Out[46]: [('Adventure,Drama,History', 7.1),
('Drama', 4.5),
('Adventure,Fantasy,Sci-Fi', 6.5),
('Action,Adventure', 5.7),
('Comedy', 6.6)]
```

```
In [58]: Avg_q9 = q9.mapValues(lambda x: (x,1))
Avg_q9 = Avg_q9.reduceByKey(lambda x,y: (x[0]+y[0], x[1]+y[1])).mapValues(lambda
da x: x[0]/x[1]).sortBy(lambda x: x[1], False)
Avg_q9.take(1)
```

```
Out[47]: [('Music,Musical', 8.5)]
```

Q10 Movies can have multiple genres. What is **the individual genre** which is the most popular as measured by number of votes. Votes for multiple genres count towards each genre listed.

Hint: Think about the wordcount exercise we have done with RDDs.

```
In [60]: q10 = movie.filter(lambda x: (x[0] == 'movie')).map(lambda x: (x[1],x[3]))
q10.take(5)
```

```
Out[48]: [('Drama', 17),
('Drama', 12),
('\N', 13),
('\N', 17),
('Crime,Thriller', 12)]
```

```
In [61]: rdd10_split = q10.map(lambda x: (x[0].split(","), x[1]))
rdd10_map = rdd10_split.flatMap(lambda x: [(y, int(x[1])) for y in x[0]]).redu
ceByKey(lambda x,y: x+y).sortBy(lambda x: x[1], False)
rdd10_map.take(1)
```

```
Out[89]: [('Drama', 405996518)]
```

Engineering the perfect cast

We have created a number of potential features for predicting the rating of a movie based on its cast. Use sparkML to build a simple linear model to predict the rating of a movie based on the following features:

1. The total number of movies in which the actors / actresses in the current movie have acted (based on Q3)
2. The average pagerank of the cast in each movie (based on Q5)
3. The average outDegree of the cast in each movie (based on Q6)
4. The average value for for the cast of degrees of Kevin Bacon (based on Q7).

If you were unable to generate any of these features as you could not answer the previous questions, just skip that particular feature.

You will need to create a dataframe with the required features and label. Use a pipeline to create the vectors required by sparkML and apply the model. Remember to split your dataset, leave 30% of the data for testing, when splitting your data use the option `seed=0`.

Q11 Provide the coefficients of the regression and the accuracy of your model on the test dataset according to RSME.

```
In [63]: # Question 3 (SUM)
from pyspark.sql.types import *
from pyspark.sql.types import StructField

df_clean = df_final.map(lambda x: (x[0][0],x[0][1],x[1]))
```

```
Out[92]: [('nm0103977', 'Brahmanandam', 809),
 ('nm0007123', 'Mammootty', 379),
 ('nm0482320', 'Mohanlal', 343),
 ('nm0149822', 'Mithun Chakraborty', 332),
 ('nm0007106', 'Shakti Kapoor', 309),
 ('nm0415549', 'Jagathi Sreekumar', 303),
 ('nm0035067', 'Cüneyt Arkin', 294),
 ('nm0374974', 'Helen', 281),
 ('nm0534867', 'Madhu', 277),
 ('nm0004429', 'Dharmendra', 270),
 ('nm0000616', 'Eric Roberts', 256),
 ('nm0045119', 'Aruna Irani', 248),
 ('nm1894124', 'Seiji Nakamitsu', 245),
 ('nm0154164', 'Soumitra Chatterjee', 244),
 ('nm0453520', 'Ji-mee Kim', 244),
 ('nm0613417', 'Raza Murad', 241),
 ('nm0474820', 'Kiran Kumar', 232),
 ('nm0315553', 'Krishna Ghattamaneni', 232),
 ('nm0893449', 'Nedumudi Venu', 230),
 ('nm0764762', 'Sharada', 229),
 ('nm0352032', 'Kamal Haasan', 227),
 ('nm1001108', 'Yuri Izumi', 225),
 ('nm0695177', 'Prakash Raj', 224),
 ('nm0595934', 'Mohan Babu', 223),
 ('nm0419707', 'Jayasudha', 218),
 ('nm0419685', 'Jaya Prada', 215),
 ('nm0993695', 'Kayoko Sugi', 214),
 ('nm0004467', 'Satyanarayana Kaikala', 211),
 ('nm0482285', 'Lakshmi', 211),
 ('nm0451600', 'Anupam Kher', 206),
 ('nm2147526', 'Asrani', 205),
 ('nm0739418', 'Gloria Romero', 204),
 ('nm0764298', 'Vilma Santos', 201),
 ('nm0159159', 'Prem Chopra', 201),
 ('nm0329730', 'Suresh Gopi', 198),
 ('nm0004109', 'Gulshan Grover', 197),
 ('nm1069583', 'Shinji Kubo', 197),
 ('nm0420090', 'Jeetendra', 193),
 ('nm0707425', 'Rajinikanth', 189),
 ('nm0004469', 'Srinivasa Rao Kota', 188),
 ('nm0158112', 'Chiranjeevi', 187),
 ('nm0154146', 'Prasenjit Chatterjee', 186),
 ('nm0621937', 'Nassar', 184),
 ('nm0000821', 'Amitabh Bachchan', 183),
 ('nm0814734', 'Türkan Soray', 182),
 ('nm0430803', 'Mohan Joshi', 181),
 ('nm0994324', 'Yutaka Ikejima', 178),
 ('nm0938860', 'Kung-won Nam', 175),
 ('nm0793731', 'Shirô Shimomoto', 173),
 ('nm0320883', 'Fatma Girik', 171),
 ('nm0349347', 'Eddie Gutierrez', 171),
```

('nm0415556', 'Jagdeep', 171),
('nm0044796', 'Raj Babbar', 170),
('nm0006763', 'Jackie Shroff', 166),
('nm0590985', 'Yûichi Minato', 166),
('nm0062540', 'Perla Bautista', 165),
('nm0707399', 'Rajendra Prasad', 165),
('nm0511276', 'Anita Linda', 164),
('nm0042124', 'Nora Aunor', 162),
('nm0042820', 'Suzan Avci', 162),
('nm0044467', 'Yuriko Azuma', 161),
('nm0802374', 'Shatrughan Sinha', 161),
('nm0044600', 'Saroja Devi B.', 160),
('nm0889148', 'Vanisri', 160),
('nm0766470', 'Sathyaraj', 160),
('nm0004334', 'Rekha', 159),
('nm0433887', 'K.R. Vijaya', 159),
('nm0083238', 'Birbal', 158),
('nm0261825', 'Joseph Estrada', 156),
('nm0000367', 'Gérard Depardieu', 154),
('nm0082848', 'Bindu', 154),
('nm0000514', 'Michael Madsen', 153),
('nm0462607', 'Hülya Koçyigit', 152),
('nm0945189', 'Simon Yam', 150),
('nm0471447', 'Ramya Krishnan', 149),
('nm0209649', 'Christopher De Leon', 148),
('nm0476429', 'Kinichi Kusumi', 142),
('nm0145061', 'Eric del Castillo', 142),
('nm0226770', 'Dileep', 141),
('nm0787462', 'Naseeruddin Shah', 141),
('nm0002002', 'Shin'ichi Chiba', 139),
('nm5083230', 'Ken'ichirô Sugiyama', 138),
('nm0004569', 'Sanjay Dutt', 137),
('nm6563624', 'Tamer Yigit', 137),
('nm0993416', 'Miki Hayashi', 135),
('nm0671381', 'Ana Luisa Peluffo', 134),
('nm1399111', 'Yôko Satomi', 133),
('nm0490489', 'Andy Lau', 133),
('nm0938893', 'Anthony Chau-Sang Wong', 132),
('nm0332871', 'Govinda', 132),
('nm0710211', 'Ranjeet', 131),
('nm0000818', 'Shabana Azmi', 130),
('nm0474774', 'Akshay Kumar', 130),
('nm0001744', 'Tom Sizemore', 129),
('nm0603865', 'Alma Moreno', 128),
('nm0626259', 'Franco Nero', 127),
('nm0733727', 'Susan Roces', 127),
('nm0417310', 'Showkar Janaki', 127),
('nm0457410', 'Ravi Kishan', 127),
('nm0066075', 'Rakesh Bedi', 127),
('nm0538690', 'Michiyo Mako', 126),
('nm0159269', 'Mohan Choti', 126),

('nm1422956', 'Kyôko Kazama', 126),
('nm0799108', 'Armando Silvestre', 126),
('nm3128033', 'Yasushi Takemoto', 126),
('nm0712546', 'Paresh Rawal', 126),
('nm0613514', 'Murali Mohan', 125),
('nm0874676', 'Eric Tsang', 125),
('nm0001803', 'Danny Trejo', 124),
('nm0007124', 'Suhasini', 124),
('nm0417270', 'Jamuna', 123),
('nm0151539', 'Chandrashekhar', 123),
('nm0261724', 'Joe Estevez', 122),
('nm0181397', 'Rez Cortez', 122),
('nm0219939', 'Danny Denzongpa', 122),
('nm0784521', 'Senthil', 122),
('nm0722029', 'Jorge Reynoso', 122),
('nm0461985', 'Akira Kobayashi', 121),
('nm1008063', 'Mimi Sawaki', 121),
('nm0619047', 'Anant Nag', 121),
('nm0030672', 'Boots Anson-Roa', 120),
('nm0408381', 'Kadir Inanir', 120),
('nm0004564', 'Hema Malini', 119),
('nm0897227', 'Vijayshanti', 118),
('nm0047962', 'Chieko Baishô', 118),
('nm0212541', 'Dipankar Dey', 118),
('nm0991856', 'Lito Lapid', 118),
('nm0015360', 'Filiz Akin', 118),
('nm0272240', 'Sevda Ferdag', 117),
('nm0498645', 'Leelavathi', 117),
('nm0317863', 'Rosemarie Gil', 117),
('nm0577281', 'Ahmet Mekin', 116),
('nm0151155', 'Michael Wai-Man Chan', 116),
('nm0084428', 'Biswajit Chatterjee', 116),
('nm0612334', 'Ranjit Mallick', 116),
('nm0012603', 'Aeng-ran Eom', 115),
('nm0999206', 'Kyôko Hashimoto', 115),
('nm0594465', 'Junko Miyashita', 113),
('nm0348029', 'Izzet Günay', 113),
('nm0001698', 'John Savage', 112),
('nm0038377', 'Ruriko Asaoka', 112),
('nm0021718', 'Fernando Almada', 112),
('nm1071598', 'Connie Chan', 112),
('nm2813324', 'Motoko Sasaki', 112),
('nm0154152', 'Sabitri Chatterjee', 111),
('nm0000366', 'Catherine Deneuve', 110),
('nm0786447', 'Gloria Sevilla', 109),
('nm1035848', 'Tomohiro Okada', 109),
('nm0013159', 'Rati Agnihotri', 109),
('nm0849437', 'Naomi Tani', 109),
('nm0155291', 'Kuan Tai Chen', 109),
('nm0906723', 'Ayako Wakao', 108),
('nm0015459', 'Nagarjuna Akkineni', 108),

('nm0830153', 'Hugo Stiglitz', 108),
('nm0219946', 'Ramesh Deo', 107),
('nm0862605', 'Kartal Tibet', 107),
('nm0949098', 'Yumi Yoshiyuki', 106),
('nm0802366', 'Mala Sinha', 106),
('nm0875460', 'Hachirô Tsuruoka', 105),
('nm0784292', 'Rituparna Sengupta', 105),
('nm0720898', 'Ramon Revilla', 104),
('nm0246150', 'Armen Dzhigarkhanyan', 104),
('nm0222426', 'Ajay Devgn', 104),
('nm0222881', 'Tony Devon', 103),
('nm1191108', 'Sulochana Latkar', 103),
('nm0811794', 'Shobana', 103),
('nm0297669', 'Hiroko Fuji', 103),
('nm0411804', 'Kazu Itsuki', 103),
('nm0851299', 'Noriko Tatsumi', 103),
('nm0438463', 'Anil Kapoor', 103),
('nm0000532', 'Malcolm McDowell', 103),
('nm0156875', 'Charlie Chin', 103),
('nm0628757', 'Francis Ng', 102),
('nm0816416', 'Murat Soydan', 102),
('nm0998011', 'Satomi Shinozaki', 101),
('nm0473314', 'Feng Ku', 101),
('nm0815018', 'Maricel Soriano', 100),
('nm0504899', 'Tony Ka Fai Leung', 100),
('nm0789374', 'Shashikala', 100),
('nm1335387', 'Prithviraj Sukumaran', 100),
('nm0734368', 'Bembol Roco', 100),
('nm0004462', 'Jean-Louis Trintignant', 100),
('nm0154139', 'Moushumi Chatterjee', 100),
('nm0874868', 'Yin Tse', 99),
('nm0000800', 'Armand Assante', 99),
('nm0000323', 'Michael Caine', 99),
('nm0256628', 'Akira Emoto', 99),
('nm0015526', 'Mehmet Ali Akpınar', 99),
('nm0347901', 'Rakhee Gulzar', 98),
('nm0001376', 'Isabelle Huppert', 98),
('nm0849863', 'Tanuja', 98),
('nm2454994', 'Bill Oberst Jr.', 98),
('nm0661262', 'Gina Pareño', 98),
('nm0348004', 'Milind Gunaji', 98),
('nm1001243', 'Kushboo', 98),
('nm0365835', 'Richard Harrison', 98),
('nm0764769', 'Ashok Saraf', 98),
('nm0794199', 'Kazuko Shirakawa', 98),
('nm0465503', 'Louis Koo', 97),
('nm0408054', 'Adel Emam', 97),
('nm0402113', 'Ediz Hun', 97),
('nm0000661', 'Donald Sutherland', 96),
('nm0796196', 'Josephine Siao', 96),
('nm0372049', 'Hotaru Hazuki', 96),

('nm0000448', 'Lance Henriksen', 96),
('nm0510857', 'Brigitte Lin', 96),
('nm1107894', 'Sivakumar', 96),
('nm2270922', 'Biswajit Chakraborty', 95),
('nm0619938', 'Tatsuya Nakadai', 95),
('nm3394756', 'Siddhanta Mahapatra', 95),
('nm0151827', 'Sylvia Chang', 95),
('nm0000640', 'Martin Sheen', 95),
('nm0862479', 'Lung Ti', 95),
('nm0000215', 'Susan Sarandon', 94),
('nm1289141', 'Lauro Delgado', 94),
('nm0001075', 'Peter Coyote', 94),
('nm0492352', 'Lan Law', 94),
('nm1241578', 'Yôta Kawase', 94),
('nm0014227', 'Sung-Ki Ahn', 94),
('nm0865994', 'Lorna Tolentino', 94),
('nm0000172', 'Harvey Keitel', 94),
('nm0000329', 'Jackie Chan', 93),
('nm0999317', 'Yukiko Tachibana', 93),
('nm0408476', 'Rafael Inclán', 93),
('nm0645327', 'Manuel Ojeda', 93),
('nm0000334', 'Yun-Fat Chow', 93),
('nm0490513', 'Ching Wan Lau', 92),
('nm3221054', 'Nadeem Baig', 92),
('nm0001424', 'Udo Kier', 92),
('nm0792911', 'Sunil Shetty', 92),
('nm0000115', 'Nicolas Cage', 92),
('nm0157747', 'Han Chin', 92),
('nm0000799', 'Edward Asner', 92),
('nm0049395', 'Nandamuri Balakrishna', 92),
('nm0000418', 'Danny Glover', 92),
('nm0004193', 'Debbie Rochon', 91),
('nm2814662', 'Kôju Ran', 91),
('nm1115537', 'Vijayakanth', 91),
('nm0920460', 'Vernon Wells', 90),
('nm0847118', 'Hideki Takahashi', 90),
('nm0000134', 'Robert De Niro', 90),
('nm0316284', 'Giancarlo Giannini', 90),
('nm0001595', 'Michael Paré', 89),
('nm1296472', 'Vic Sotto', 89),
('nm0747155', 'Reena Roy', 89),
('nm6150259', 'Bobita', 89),
('nm0612614', 'Mumtaz', 89),
('nm0004487', 'Juhi Chawla', 89),
('nm0219971', 'Sunny Deol', 88),
('nm0105475', 'Claude Brasseur', 88),
('nm0000728', 'Mario Adorf', 88),
('nm0005033', 'Sammo Kam-Bo Hung', 88),
('nm0643350', 'Suresh Oberoi', 88),
('nm0174240', 'Gabby Concepcion', 88),
('nm0066223', 'Kenny Bee', 87),

('nm0401192', 'Kara Wai', 87),
('nm0497097', 'Danny Lee', 87),
('nm0001012', 'Claudia Cardinale', 87),
('nm0156955', 'David Chiang', 87),
('nm0324845', 'Vikram Gokhale', 87),
('nm0000461', 'Michael Ironside', 86),
('nm0155587', 'Kent Cheng', 86),
('nm0297686', 'Tatsuya Fuji', 86),
('nm0874684', 'Kenneth Tsang', 86),
('nm0729473', 'Jorge Rivero', 86),
('nm0001626', 'Christopher Plummer', 86),
('nm0786928', 'Yusuf Sezgin', 86),
('nm1069551', 'Eun-a Ko', 86),
('nm0001367', 'C. Thomas Howell', 85),
('nm0848396', 'Rumi Tama', 85),
('nm0351565', 'Salih Güney', 85),
('nm0471464', 'Krishnamraju', 85),
('nm0035018', 'Arjun Sarja', 85),
('nm0544424', 'Edu Manzano', 85),
('nm0006795', 'Salman Khan', 85),
('nm0000353', 'Willem Dafoe', 84),
('nm0846616', 'Sharmila Tagore', 84),
('nm0022765', 'Héctor Alterio', 84),
('nm0244900', 'Rajatabha Dutta', 84),
('nm0244707', 'André Dussollier', 84),
('nm0594318', 'Kaoru Miya', 84),
('nm0896573', 'Ashish Vidyarthi', 84),
('nm0620699', 'Nan Chiang', 84),
('nm1383984', 'Rachana Banerjee', 84),
('nm0000168', 'Samuel L. Jackson', 84),
('nm0350884', 'Ernesto Gómez Cruz', 83),
('nm0893142', 'Venkatesh Daggubati', 83),
('nm0410902', 'Renji Ishibashi', 83),
('nm0033175', 'Arathi', 83),
('nm0720763', 'Revathy', 83),
('nm0645422', 'Mariko Okada', 83),
('nm0001036', 'Geraldine Chaplin', 83),
('nm0000553', 'Liam Neeson', 82),
('nm0959872', 'Nebahat Çehre', 82),
('nm0023868', 'Zeenat Aman', 81),
('nm0000809', 'Daniel Auteuil', 81),
('nm1187366', 'Mayuko Sasaki', 81),
('nm0396212', 'Yukijirô Hotaru', 81),
('nm1027829', 'Tarô Araki', 81),
('nm0820241', 'Srikanth', 81),
('nm0416077', 'Farida Jalal', 81),
('nm0047016', 'Ying Bai', 80),
('nm0396069', 'Hassan Hosny', 80),
('nm2852415', 'Mami Sakura', 80),
('nm0001136', 'Bruce Dern', 80),
('nm0949045', 'Sayuri Yoshinaga', 80),

('nm0953881', 'Alfonso Zayas', 80),
('nm0628806', 'Man-Tat Ng', 80),
('nm0406393', 'Manuel 'Flaco' Ibáñez', 79),
('nm0451425', 'Kulbhushan Kharbanda', 79),
('nm0000929', 'Corbin Bernsen', 79),
('nm0442207', 'Lloyd Kaufman', 79),
('nm0038355', 'Tadanobu Asano', 79),
('nm0149837', 'Sabyasachi Chakrabarty', 79),
('nm0284533', 'Alex Fong', 79),
('nm0560962', 'Carmen Maura', 78),
('nm0412615', 'Shima Iwashita', 78),
('nm0001426', 'Ben Kingsley', 78),
('nm0095108', 'César Bono', 78),
('nm0848993', 'Kunie Tanaka', 78),
('nm0505323', 'Johnny Lever', 78),
('nm0755364', 'José Sacristán', 78),
('nm0510950', 'Sandra Kwan Yue Ng', 78),
('nm0271826', 'Feng-Jiao Lin', 78),
('nm0683298', 'Pilar Pilapil', 78),
('nm0620700', 'Hong Nan', 77),
('nm0083534', 'Hemant Birje', 77),
('nm1835962', 'Sanae Shiba', 77),
('nm0661239', 'Asha Parekh', 77),
('nm0000246', 'Bruce Willis', 77),
('nm0039593', 'Usagi Asô', 77),
('nm0868381', 'Joel Torre', 77),
('nm0442398', 'Satish Kaul', 77),
('nm0875362', 'Yôko Tsukasa', 77),
('nm0151866', 'Yi Chang', 77),
('nm0945475', 'Ryûji Yamamoto', 77),
('nm0656865', 'Paintal', 77),
('nm0628756', 'Fung Woo', 76),
('nm0596807', 'Ángela Molina', 76),
('nm2278431', 'Joe Hammerstone', 76),
('nm0437147', 'Kanchana', 76),
('nm0451387', 'Padma Khanna', 76),
('nm0622186', 'Alok Nath', 76),
('nm0003909', 'Michael Lonsdale', 76),
('nm0043956', 'Nevin Aypar', 75),
('nm0000164', 'Anthony Hopkins', 75),
('nm0463539', 'Manisha Koirala', 75),
('nm0847361', 'Akira Takarada', 75),
('nm1031561', 'Rie Nakano', 75),
('nm0059631', 'Çetin Basaran', 75),
('nm0001128', 'Alain Delon', 75),
('nm0001041', 'Maggie Cheung', 75),
('nm2884462', 'Riri Kôda', 74),
('nm0784884', 'Rade Serbedzija', 74),
('nm0504973', 'Alex Man', 74),
('nm0007102', 'Tabu', 74),
('nm0051856', 'Victor Banerjee', 74),

('nm0000104', 'Antonio Banderas', 74),
('nm0004365', 'Fred Williamson', 74),
('nm0149816', 'Chiranjit', 74),
('nm0810342', 'William Smith', 74),
('nm2671390', 'Alamgir', 73),
('nm0438092', 'Dimple Kapadia', 73),
('nm0751638', 'James Russo', 73),
('nm0504897', 'Tony Chiu-Wai Leung', 73),
('nm0645382', 'Naomi Oka', 73),
('nm0000708', 'Billy Zane', 73),
('nm0202966', 'Keith David', 73),
('nm0297670', 'Sumiko Fuji', 73),
('nm0042564', 'Amy Austria', 73),
('nm0157971', 'Paul Chun', 73),
('nm0290556', 'James Franco', 73),
('nm0704157', 'Guillermo Quintanilla', 73),
('nm0264303', 'Moeko Ezawa', 72),
('nm0475610', 'Leonid Kuravlyov', 72),
('nm0435356', 'Costas Kakavas', 72),
('nm0000151', 'Morgan Freeman', 72),
('nm0092184', 'Richard Bohringer', 72),
('nm0038960', 'Izumi Ashikawa', 72),
('nm3492497', 'Shakib Khan', 72),
('nm0716851', 'Waheeda Rehman', 71),
('nm0788152', 'Shanker', 71),
('nm0747178', 'Sandhya Roy', 71),
('nm0792116', 'Jimmy Sheirgill', 71),
('nm0000721', 'Victoria Abril', 71),
('nm0559385', 'Chieko Matsubara', 71),
('nm0008346', 'Hiroshi Abe', 71),
('nm0311497', 'Ihsan Gedik', 71),
('nm0014558', 'Shô Aikawa', 71),
('nm0211565', 'Christian De Sica', 71),
('nm0000313', 'Jeff Bridges', 71),
('nm0305955', 'Andrés García', 70),
('nm0795517', 'Qi Shu', 70),
('nm0015142', 'Aydemir Akbas', 70),
('nm1142519', 'Ramon 'Bong' Revilla Jr.', 70),
('nm0288206', 'Nagwa Fouad', 70),
('nm0768334', 'John Saxon', 70),
('nm1202543', 'Efren Reyes Jr.', 70),
('nm0645550', 'Arzu Okay', 70),
('nm0000476', 'Sally Kirkland', 70),
('nm0007113', 'Nana Patekar', 70),
('nm0434593', 'Kyôko Kagawa', 70),
('nm0150952', 'Jordan Chan', 70),
('nm0144589', 'Lou Castel', 70),
('nm0038167', 'Jamshid Hashempur', 70),
('nm0000598', 'Dennis Quaid', 70),
('nm0849199', 'Raveena Tandon', 70),
('nm0646037', 'Daniel Olbrychski', 70),

('nm0000432', 'Gene Hackman', 69),
('nm0911093', 'Jimmy Wang Yu', 69),
('nm0000997', 'Gary Busey', 69),
('nm0002043', 'Madhuri Dixit', 69),
('nm0000554', 'Sam Neill', 69),
('nm0828288', 'Brinke Stevens', 69),
('nm0327076', 'Richard Gomez', 69),
('nm0001560', 'Ornella Muti', 69),
('nm0992865', 'Aki Izumi', 69),
('nm0351422', 'Melek Görgün', 68),
('nm0624075', 'Ni Tien', 68),
('nm0348122', 'Selma Güneri', 68),
('nm0001002', 'Dean Cain', 68),
('nm0006573', 'Philippe Leroy', 68),
('nm0000185', 'Dolph Lundgren', 68),
('nm0019815', 'Selda Alkor', 68),
('nm0865302', 'Tony Todd', 68),
('nm0095007', 'Dina Bonnevie', 68),
('nm0741378', 'Felissa Rose', 68),
('nm0971442', 'Helen Gamboa', 68),
('nm1256435', 'Dündar Aydinli', 68),
('nm0043708', 'Aynur Aydan', 68),
('nm0000901', 'Jean-Paul Belmondo', 68),
('nm0298689', 'Stanley Sui-Fan Fung', 68),
('nm0001001', 'James Caan', 67),
('nm0908914', 'Dee Wallace', 67),
('nm0493719', 'Norma Lazareno', 67),
('nm0950690', 'Yang Yueh', 67),
('nm0933727', 'Lambert Wilson', 67),
('nm0244333', 'Ejaz Durrani', 67),
('nm0879556', 'Mari Töröcsik', 67),
('nm0746987', 'Debashree Roy', 67),
('nm0046894', 'Mohnish Bahl', 67),
('nm0659156', 'Aditya Pancholi', 67),
('nm0521808', 'Simon Lui', 67),
('nm4025258', 'Setsu Shimizu', 67),
('nm0004363', 'Rakesh Roshan', 67),
('nm0006939', 'Richard Berry', 67),
('nm0611644', 'Niño Muhlach', 67),
('nm0474609', 'Atul Kulkarni', 67),
('nm0556319', 'Kyôsukey Machida', 67),
('nm0001648', 'Charlotte Rampling', 67),
('nm1699511', 'Momoyo Ôkawa', 66),
('nm0000407', 'Vivica A. Fox', 66),
('nm0790659', 'Sheela', 66),
('nm0000579', 'Ron Perlman', 66),
('nm0201669', 'Jean-Pierre Darroussin', 66),
('nm0008206', 'Akbar Abdi', 66),
('nm0160865', 'Norman Chu', 66),
('nm0762248', 'Stefania Sandrelli', 66),
('nm0209660', 'Joey de Leon', 66),

('nm0155599', 'Mark Cheng', 66),
('nm0023832', 'Mathieu Amalric', 66),
('nm1007226', 'Mayuko Hino', 66),
('nm0000285', 'Alec Baldwin', 66),
('nm0048075', 'Manoj Bajpayee', 65),
('nm0197582', 'Sudhir Dalvi', 65),
('nm0000047', 'Sophia Loren', 65),
('nm0000919', 'Senta Berger', 65),
('nm0722636', 'John Rhys-Davies', 65),
('nm0701121', 'Esen Püsküllü', 65),
('nm0595672', 'Bahman Mofid', 65),
('nm1521381', 'Nayanthara', 65),
('nm0001117', 'Bruce Davison', 65),
('nm0396136', 'Robert Hossein', 65),
('nm0001166', 'James Duval', 65),
('nm3845073', 'Sabiha Khanum', 65),
('nm0000131', 'John Cusack', 65),
('nm0037593', 'Göksel Arsoy', 64),
('nm0247982', 'Nabila Ebeid', 64),
('nm0950350', 'Hasan Youssef', 64),
('nm0000374', 'Brad Dourif', 64),
('nm0167388', 'François Cluzet', 64),
('nm0696163', 'Micheline Presle', 64),
('nm0913911', 'Tetsuya Watari', 64),
('nm0000874', 'Steven Bauer', 64),
('nm0708095', 'Rambha', 64),
('nm0628731', 'Carrie Ng', 64),
('nm0066827', 'Bahar Begum', 64),
('nm0000321', 'Gabriel Byrne', 64),
('nm0836979', 'Kemi Ichiboshi', 64),
('nm0001745', 'Stellan Skarsgård', 64),
('nm0000420', 'Valeria Golino', 64),
('nm0430482', 'Jaclyn Jose', 64),
('nm0051536', 'Pouri Baneai', 64),
('nm0683831', 'Silvia Pinal', 64),
('nm0451307', 'Saif Ali Khan', 63),
('nm0190691', 'György Cserhalmi', 63),
('nm0451321', 'Shah Rukh Khan', 63),
('nm0523344', 'Lynn Lowry', 63),
('nm0000882', 'Nathalie Baye', 63),
('nm0611538', 'Madhabi Mukherjee', 63),
('nm0007746', 'Diego Abatantuono', 63),
('nm0246640', 'Julio Diaz', 63),
('nm0050197', 'Yogeeta Bali', 63),
('nm0000518', 'John Malkovich', 63),
('nm0012881', 'Mohan Agashe', 63),
('nm0000838', 'Daniel Baldwin', 63),
('nm0508293', 'Thierry Lhermitte', 63),
('nm0904537', 'Vyjayanthimala', 63),
('nm0246582', 'Gloria Diaz', 63),
('nm0332709', 'Olivier Gourmet', 63),

('nm0067808', 'Fatma Belgen', 63),
('nm0155562', 'Ekin Cheng', 63),
('nm0892260', 'Concha Velasco', 63),
('nm0955603', 'Zhen Zhen', 63),
('nm0271787', 'Bo-Bo Fung', 63),
('nm1913625', 'Parambrata Chattopadhyay', 63),
('nm3025400', 'Shawn C. Phillips', 63),
('nm1156207', 'Riaz', 63),
('nm0149846', 'Lily Chakravarty', 63),
('nm0256861', 'Ken'ichi Endô', 63),
('nm1031338', 'Aya Midorikawa', 63),
('nm0001194', 'Jeff Fahey', 63),
('nm0523791', 'Antonella Lualdi', 62),
('nm0150850', 'Pak-Cheung Chan', 62),
('nm3981484', 'Asif Ali', 62),
('nm0000603', 'Vanessa Redgrave', 62),
('nm0066455', 'Beena Banerjee', 62),
('nm1665526', 'Mahmoud Yassine', 62),
('nm1679372', 'Sudeep', 62),
('nm0207338', 'Janice de Belen', 62),
('nm0000380', 'Robert Duvall', 62),
('nm0801264', 'Simran', 62),
('nm0000191', 'Ewan McGregor', 62),
('nm0151860', 'Yang Chang', 62),
('nm5083762', 'Bun'ei Shô', 61),
('nm0000174', 'Val Kilmer', 61),
('nm0767491', 'Perihan Savas', 61),
('nm0398884', 'Chin Hu', 61),
('nm0001285', 'Elliott Gould', 61),
('nm0000173', 'Nicole Kidman', 61),
('nm0000686', 'Christopher Walken', 61),
('nm0583951', 'Robert Miano', 61),
('nm0271763', 'Edwige Fenech', 61),
('nm0000606', 'Jean Reno', 61),
('nm0594257', 'Tomokazu Miura', 61),
('nm0661886', 'Geun-hyeong Park', 61),
('nm0000437', 'Woody Harrelson', 61),
('nm0686375', 'Michele Placido', 61),
('nm0081175', 'Jean-Luc Bideau', 61),
('nm0879186', 'Richard Tyson', 61),
('nm1375534', 'Trisha Krishnan', 61),
('nm0993576', 'Kanao Kishi', 61),
('nm0000194', 'Julianne Moore', 61),
('nm0000545', 'Helen Mirren', 60),
('nm0297754', 'Jun Fujimaki', 60),
('nm0000945', 'Jane Birkin', 60),
('nm0001845', 'Forest Whitaker', 60),
('nm0477209', 'Aaron Kwok', 60),
('nm0466581', 'Hilda Koronel', 60),
('nm0891835', 'Isela Vega', 60),
('nm0422941', 'Flor Silvestre', 60),

('nm0000499', 'Bai Ling', 60),
('nm0223563', 'Poonam Dhillon', 60),
('nm0000491', 'John Leguizamo', 60),
('nm0000920', 'Patrick Bergin', 60),
('nm0210218', 'Maria de Medeiros', 60),
('nm0474801', 'Dilip Kumar', 60),
('nm0603090', 'Laura Morante', 60),
('nm0043711', 'Nilüfer Aydan', 60),
('nm0184392', 'Martin Kove', 60),
('nm0025627', 'Tinnu Anand', 60),
('nm0156484', 'Jacky Cheung', 60),
('nm0001409', 'Tchéky Karyo', 60),
('nm0000300', 'Juliette Binoche', 60),
('nm0443232', 'Yûzô Kayama', 60),
('nm0352157', 'Alessandro Haber', 60),
('nm0244890', 'Divya Dutta', 60),
('nm0192117', 'Sharon Cuneta', 60),
('nm0000438', 'Ed Harris', 60),
('nm0847562', 'Naoto Takenaka', 60),
('nm1493417', 'Yun Ling', 60),
('nm0897201', 'Joseph Vijay', 60),
('nm0619185', 'Masatoshi Nagase', 60),
('nm0001757', 'Kevin Sorbo', 59),
('nm0125627', 'Lando Buzzanca', 59),
('nm0639152', 'Ghita Nørby', 59),
('nm0477093', 'Chun Yang', 59),
('nm0000272', 'Fanny Ardant', 59),
('nm1001584', 'Reiko Ôtsuki', 59),
('nm0000286', 'Stephen Baldwin', 59),
('nm0593045', 'Yoshiko Mita', 59),
('nm0000483', 'Christopher Lambert', 59),
('nm0000502', 'Christopher Lloyd', 59),
('nm0045136', 'Sarika', 59),
('nm0024910', 'Mervat Amin', 59),
('nm0497394', 'Heung Kam Lee', 59),
('nm0368990', 'Rohini Hattangadi', 59),
('nm0001108', 'Robert Davi', 59),
('nm0000125', 'Sean Connery', 59),
('nm0591877', 'Miou-Miou', 59),
('nm0005078', 'Stacy Keach', 59),
('nm0000458', 'William Hurt', 59),
('nm1056425', 'Rajpal Yadav', 59),
('nm0021835', 'Joaquim de Almeida', 59),
('nm0949350', 'Burt Young', 59),
('nm0434628', 'Hidetoshi Kageyama', 59),
('nm0622732', 'Deepti Naval', 59),
('nm0201000', 'Hülya Darcı', 59),
('nm1002952', 'Yuka Asagiri', 59),
('nm0697514', 'Barry Prima', 59),
('nm0000658', 'Meryl Streep', 59),
('nm0315528', 'Faramarz Gharibian', 58),

('nm0599004', 'Cesar Montano', 58),
('nm0156533', 'Nick Cheung', 58),
('nm0116254', 'Valeria Bruni Tedeschi', 58),
('nm0900557', 'Marina Vlady', 58),
('nm0000707', 'Sean Young', 58),
('nm0794827', 'Kuan-Hsiung Wang', 58),
('nm0000422', 'John Goodman', 58),
('nm0001283', 'Louis Gossett Jr.', 58),
('nm0595115', 'W.D. Mochtar', 58),
('nm0206870', 'Luis de Alba', 58),
('nm0475602', 'Yasuaki Kurata', 58),
('nm0006433', 'Karisma Kapoor', 58),
('nm0541576', 'Nick Mancuso', 58),
('nm0802251', 'Upasna Singh', 58),
('nm0032661', 'Müjde Ar', 58),
('nm0000354', 'Matt Damon', 58),
('nm0792866', 'Meenakshi Sheshadri', 58),
('nm0465062', 'Tao Chiang', 58),
('nm0541908', 'Costas Mandylor', 58),
('nm0161275', 'Siu Chung Mok', 58),
('nm0000206', 'Keanu Reeves', 58),
('nm0758345', 'Carmen Salinas', 58),
('nm0000218', 'Kristin Scott Thomas', 58),
('nm0737730', 'Roja', 58),
('nm0044481', 'Yôko Azusa', 58),
('nm0001929', 'Josiane Balasko', 58),
('nm0328112', 'Dacia González', 57),
('nm0939255', 'Michael Wong', 57),
('nm0498189', 'Sam Lee', 57),
('nm0000663', 'Dominique Swain', 57),
('nm0897628', 'Lucha Villa', 57),
('nm0000158', 'Tom Hanks', 57),
('nm0766233', 'Kôichi Satô', 57),
('nm0411539', 'Puneet Issar', 57),
('nm0351366', 'Serdar Gökhan', 57),
('nm1384413', 'Saswata Chatterjee', 57),
('nm0756378', 'Parikshit Sahni', 57),
('nm0001235', 'William Forsythe', 57),
('nm0284717', 'Malini Fonseka', 57),
('nm1961459', 'Tamannaah Bhatia', 57),
('nm0015001', 'Ajith Kumar', 57),
('nm0149008', 'Martha Elena Cervantes', 57),
('nm1307939', 'Meera Jasmine', 57),
('nm0224933', 'Chryssoula Diavati', 57),
('nm0349427', 'Emilio Gutiérrez Caba', 57),
('nm4468244', 'Arun Govil', 56),
('nm0620241', 'Tamao Nakamura', 56),
('nm0893941', 'Maribel Verdú', 56),
('nm1923635', 'Kikujirô Honda', 56),
('nm0945599', 'Yuri Yamashina', 56),
('nm0157785', 'Siu-Ho Chin', 56),

('nm0001934', 'Kabir Bedi', 56),
('nm0000302', 'Jacqueline Bisset', 56),
('nm0442470', 'Kamini Kaushal', 56),
('nm0892816', 'Lorena Velázquez', 56),
('nm0073031', 'Femi Benussi', 56),
('nm0000230', 'Sylvester Stallone', 56),
('nm0252309', 'Ali Ekdal', 56),
('nm0000225', 'Christian Slater', 56),
('nm0674742', 'Jacques Perrin', 56),
('nm0000090', 'Armin Mueller-Stahl', 56),
('nm0075650', 'Charles Berling', 56),
('nm0895759', 'Karin Viard', 56),
('nm1383799', 'Ferdous Ahmed', 56),
('nm0013039', 'Morteza Aghili', 56),
('nm0000511', 'Shirley MacLaine', 56),
('nm0075710', 'François Berléand', 56),
('nm2186174', 'Feng Chang', 56),
('nm0351674', 'Nedret Güvenç', 56),
('nm0013833', 'Ah-Lei Gua', 56),
('nm0998044', 'Sneha', 56),
('nm0001505', 'Joe Mantegna', 55),
('nm0938975', 'Carter Wong', 55),
('nm4535518', 'Mindy Robinson', 55),
('nm0155607', 'Pei-Pei Cheng', 55),
('nm0464075', 'Padmini Kolhapure', 55),
('nm0351673', 'Sezer Güvenirgil', 55),
('nm0632664', 'Toshiyuki Nishida', 55),
('nm0694066', 'Clifton Powell', 55),
('nm0856500', 'Sylvie Testud', 55),
('nm0611285', 'Aga Muhlach', 55),
('nm3148014', 'Dawna Lee Heising', 55),
('nm0000460', 'Jeremy Irons', 55),
('nm0000237', 'John Travolta', 55),
('nm0875416', 'Hua Tsung', 55),
('nm0000620', 'Mickey Rourke', 55),
('nm0051880', 'Lino Banfi', 55),
('nm0559698', 'Keiko Matsuzaka', 55),
('nm0000649', 'Paul Sorvino', 55),
('nm0000560', 'Nick Nolte', 55),
('nm5954636', 'Sergey A.', 55),
('nm0387987', 'Kane Hodder', 55),
('nm0530365', 'Sergi López', 55),
('nm0039591', 'Kumiko Asô', 55),
('nm0007069', 'Pierre Richard', 55),
('nm0015099', 'Aynur Akarsu', 55),
('nm0415777', 'Ka-Yan Leung', 55),
('nm0000297', 'Tom Berenger', 55),
('nm0000619', 'Tim Roth', 55),
('nm0786443', 'Carmen Sevilla', 55),
('nm0619324', 'Nagma', 55),
('nm0099677', 'Michel Bouquet', 55),

('nm0000136', 'Johnny Depp', 55),
('nm0939153', 'Joey Wang', 55),
('nm0293739', 'Sami Frey', 55),
('nm0543547', 'Predrag 'Miki' Manojlovic", 55),
('nm0159337', 'Kabori Sarwar', 55),
('nm2421786', 'Takahiro Nomura', 55),
('nm2023617', 'Sha-Li Chen', 54),
('nm0813961', 'Elke Sommer', 54),
('nm0156522', 'Man Cheung', 54),
('nm0673449', 'Barbara Perez', 54),
('nm0031967', 'Aparna Sen', 54),
('nm0017343', 'Damián Alcázar', 54),
('nm0644680', 'Bulle Ogier', 54),
('nm0000112', 'Pierce Brosnan', 54),
('nm0694843', 'Renato Pozzetto', 54),
('nm0047963', 'Mitsuko Baishô', 54),
('nm0720277', 'Antonio Resines', 54),
('nm1851431', 'Yuen Kao', 54),
('nm0842770', 'Tilda Swinton', 54),
('nm0001643', 'Linnea Quigley', 54),
('nm0028487', 'Simón Andreu', 54),
('nm0862882', 'Tien Niu', 54),
('nm0000160', 'Ethan Hawke', 54),
('nm3056725', 'Paran Banerjee', 54),
('nm9845146', 'Uttar Kumar', 54),
('nm0513298', 'Helga Liné', 54),
('nm0140649', 'Mathieu Carrière', 54),
('nm1085810', 'Hsiao Pao Ko', 54),
('nm0903750', 'Behrouz Vossoughi', 54),
('nm3462447', 'Shivarajkumar', 54),
('nm1011348', 'Indrajith Sukumaran', 54),
('nm0473541', 'Naoko Kubo', 54),
('nm0497710', 'Kyeong-yeong Lee', 54),
('nm0707268', 'Anita Raj', 54),
('nm0000872', 'Patrick Bauchau', 53),
('nm0764156', 'Judy Ann Santos', 53),
('nm0993544', 'Etsuko Hara', 53),
('nm0846630', 'Tomorô Taguchi', 53),
('nm0602976', 'Sofia Moran', 53),
('nm0333088', 'Sergio Goyri', 53),
('nm0000995', 'Ellen Burstyn', 53),
('nm0005351', 'Ryan Reynolds', 53),
('nm1974249', 'Qi Fu', 53),
('nm0000274', 'David Arquette', 53),
('nm0000949', 'Cate Blanchett', 53),
('nm1473166', 'Amin Hayayee', 53),
('nm0357264', 'Mitsuo Hamada', 53),
('nm0709359', 'Ashutosh Rana', 53),
('nm1591928', 'Thomas Goersch', 53),
('nm0000602', 'Robert Redford', 53),
('nm1001840', 'Hiromi Saotome', 53),

('nm0004851', 'Penélope Cruz', 53),
('nm0582378', 'Sombat Metanee', 53),
('nm0898913', 'Vineeth', 53),
('nm0006764', 'Sonali Kulkarni', 53),
('nm0250808', 'Zerrin Egeliler', 53),
('nm0945131', 'Kôji Yakusho', 53),
('nm0562210', 'Antonio Mayans', 53),
('nm0001058', 'Joan Collins', 53),
('nm0836681', 'Jean-François Stévenin', 53),
('nm1094188', 'Rohini', 53),
('nm1107053', 'Robin Padilla', 53),
('nm1763351', 'Chung Chow', 53),
('nm0631963', 'Sergey Nikonenko', 53),
('nm0000352', 'Vincent D'Onofrio', 53),
('nm0490500', 'Carina Lau', 53),
('nm0514998', 'Ray Lui', 53),
('nm0214240', 'Deeba Begum', 53),
('nm0939429', 'Jung Wang', 53),
('nm0034836', 'Ineko Arima', 52),
('nm0209349', 'Eduardo de la Peña', 52),
('nm0004051', 'Brian Cox', 52),
('nm0271829', 'Nu Fenghuang', 52),
('nm0000244', 'Sigourney Weaver', 52),
('nm0156228', 'Patrick Chesnais', 52),
('nm0099054', 'Barbara Bouchet', 52),
('nm0875275', 'Elvis Tsui', 52),
('nm0000142', 'Clint Eastwood', 52),
('nm0000473', 'Diane Keaton', 52),
('nm0512071', 'Vincent Lindon', 52),
('nm0000662', 'Kiefer Sutherland', 52),
('nm0451561', 'Sachin Khedekar', 52),
('nm0482695', 'Suet Lam', 52),
('nm0001777', 'Dean Stockwell', 52),
('nm1007928', 'Mat Ranillo III', 52),
('nm0534856', 'Madhavan', 52),
('nm0473228', 'Hardy Krüger', 52),
('nm0000501', 'Ray Liotta', 52),
('nm0671831', 'Peng Tien', 52),
('nm1231899', 'Priyanka Chopra', 52),
('nm0001868', 'Michael York', 52),
('nm0283754', 'Spyros Fokas', 52),
('nm0529543', 'Jean-Pierre Léaud', 52),
('nm0000375', 'Robert Downey Jr.', 52),
('nm0225055', 'Vic Diaz', 52),
('nm0001057', 'Toni Collette', 52),
('nm0066093', 'Ahmad Bedair', 52),
('nm0080180', 'Bhagyaraj', 52),
('nm0457554', 'Sakae Nitta', 52),
('nm0485707', 'Hsiu-Shen Liang', 52),
('nm0034079', 'Pierre Arditi', 52),
('nm0297788', 'Shiho Fujimura', 52),

('nm0001218', 'Sean Patrick Flanery', 52),
('nm0225921', 'Juan Diego', 52),
('nm0943079', 'Daniel Wu', 52),
('nm0307628', 'Sergey Garmash', 52),
('nm2824472', 'Cecilia Lopez', 51),
('nm0107012', 'Jana Brejchová', 51),
('nm0000147', 'Colin Firth', 51),
('nm0001686', 'Cynthia Rothrock', 51),
('nm0534858', 'Kavya Madhavan', 51),
('nm0000276', 'Sean Astin', 51),
('nm0514904', 'Chia-Hui Liu', 51),
('nm1421465', 'Priyamani', 51),
('nm0000198', 'Gary Oldman', 51),
('nm0000412', 'Andy Garcia', 51),
('nm0000154', 'Mel Gibson', 51),
('nm0319834', 'Teresa Gimpera', 51),
('nm0947236', 'Mikhail Efremov', 51),
('nm0263851', 'Rosamund Kwan', 51),
('nm0094789', 'Sandrine Bonnaire', 51),
('nm0149247', 'Youssef Chaban', 51),
('nm0125540', 'Margherita Buy', 51),
('nm0422586', 'Ayesha Jhulka', 51),
('nm0960013', 'Yukari Ôshima', 51),
('nm0947447', 'Donnie Yen', 51),
('nm0375787', 'Aziza Helmy', 51),
('nm0757677', 'Carlos Salazar', 51),
('nm0497782', 'Loletta Lee', 51),
('nm0497763', 'Lily Li', 51),
('nm0040545', 'Féodor Atkine', 50),
('nm0784025', 'Raima Sen', 50),
('nm0481737', 'Yiu-Cheung Lai', 50),
('nm0275138', 'Andréa Ferréol', 50),
('nm0000152', 'Richard Gere', 50),
('nm0000146', 'Ralph Fiennes', 50),
('nm0997616', 'Akemi Nijô', 50),
('nm0345999', 'Blanca Guerra', 50),
('nm0001993', 'Vincent Cassel', 50),
('nm0001159', 'Faye Dunaway', 50),
('nm0007107', 'Urmila Matondkar', 50),
('nm0287891', 'Cihangir Gaffari', 50),
('nm0399007', 'Jung-Lee Hwang', 50),
('nm0536095', 'Benoît Magimel', 50),
('nm0498429', 'Waise Lee', 50),
('nm0000227', 'Mira Sorvino', 50),
('nm1397299', 'Shakuntala Barua', 50),
('nm2766218', 'Julie Anne Prescott', 50),
('nm0000169', 'Tommy Lee Jones', 50),
('nm0757293', 'Yoshiko Sakuma', 50),
('nm1303433', 'John Abraham', 50),
('nm1970665', 'Mutsuo Yoshioka', 50),
('nm0000210', 'Julia Roberts', 50),

('nm0846681', 'Dalip Tahil', 50),
('nm2570245', 'Kajal Aggarwal', 50),
('nm0473984', 'Yoshiko Kuga', 50),
('nm0043362', 'Hülya Avsar', 50),
('nm0056817', 'Bessie Barredo', 50),
('nm0000961', 'Timothy Bottoms', 50),
('nm0084443', 'Seema Biswas', 50),
('nm0119069', 'Juozas Budraitis', 50),
('nm0398904', 'Sibelle Hu', 50),
('nm0784018', 'Moon Moon Sen', 50),
('nm1946407', 'Kay Kay Menon', 50),
('nm0000242', 'Mark Wahlberg', 50),
('nm1395383', 'Mila del Sol', 50),
('nm0437156', 'Ilias Kanchan', 50),
('nm0264554', 'Françoise Fabian', 50),
('nm0287471', 'Mohammad Reza Forutan', 50),
('nm0557609', 'Valerio Mastandrea', 50),
('nm0002071', 'Will Ferrell', 50),
('nm0222983', 'Devlet Devrim', 50),
('nm1044659', 'Boy Alano', 50),
('nm0412819', 'Masako Izumi', 50),
('nm0159507', 'Stephen Chow', 50),
('nm0508085', 'Johan Leysen', 50),
('nm0001804', 'Stanley Tucci', 50),
('nm0015440', 'Kumiko Akiyoshi', 50),
('nm0000621', 'Kurt Russell', 50),
('nm0734558', 'Karel Roden', 50),
('nm0614971', 'Guillermo Murray', 49),
('nm0180404', 'Clovis Cornillac', 49),
('nm0155546', 'Carol "Do Do" Cheng', 49),
('nm0001638', 'Jürgen Prochnow', 49),
('nm1471547', 'Mahnaz Afshar', 49),
('nm0623387', 'Yuriy Nazarov', 49),
('nm1417314', 'Vikram', 49),
('nm0628827', 'Richard Ng', 49),
('nm0151534', 'Sudha Chandran', 49),
('nm0000546', 'Matthew Modine', 49),
('nm0000733', 'Anouk Aimée', 49),
('nm0623164', 'Takashi Naha', 49),
('nm0502425', 'Melissa Leo', 49),
('nm0001831', 'David Warner', 49),
('nm0000991', 'Geneviève Bujold', 49),
('nm0954704', 'Roschdy Zem', 49),
('nm0006370', 'Manoj Kumar', 49),
('nm0000551', 'Dermot Mulroney', 49),
('nm0487254', 'Gérard Lanvin', 49),
('nm0915208', 'Naomi Watts', 49),
('nm0524528', 'Fabrice Luchini', 49),
('nm0000126', 'Kevin Costner', 49),
('nm0273646', 'Maribel Fernández', 49),
('nm0945734', 'Tsutomu Yamazaki', 49),

('nm0437630', 'Kar-Ying Law', 49),
('nm0036734', 'Françoise Arnoul', 49),
('nm1294464', 'Delia Razon', 49),
('nm0712433', 'Ravi Teja', 49),
('nm0518178', 'Gina Lollobrigida', 49),
('nm0462685', 'Maro Kodou', 49),
('nm0092789', 'Massimo Boldi', 49),
('nm1127958', 'Yograj Singh', 49),
('nm0027683', 'Harriet Andersson', 49),
('nm0512689', 'Ivy Ling Po', 49),
('nm0000102', 'Kevin Bacon', 49),
('nm0157739', 'Feng Chin', 49),
('nm0000547', 'Alfred Molina', 49),
('nm0000232', 'Sharon Stone', 48),
('nm0001352', 'Terence Hill', 48),
('nm0441104', 'Yuko Katagiri', 48),
('nm0072768', 'Fabrizio Bentivoglio', 48),
('nm1078422', 'Hirofumi Arai', 48),
('nm0004626', 'Kareena Kapoor', 48),
('nm0442955', 'Yûsuke Kawazu', 48),
('nm0829155', 'Alexandra Stewart', 48),
('nm1249052', 'Riccardo Scamarcio', 48),
('nm0894969', 'Gardo Versoza', 48),
('nm0439312', 'Niki Karimi', 48),
('nm0764724', 'Diler Saraç', 48),
('nm0150921', 'Hui Lou Chen', 48),
('nm0086926', 'Erika Blanc', 48),
('nm0000197', 'Jack Nicholson', 48),
('nm0790863', 'Javed Sheikh', 48),
('nm0000335', 'Glenn Close', 48),
('nm0760796', 'Hiroyuki Sanada', 48),
('nm0000299', 'Michael Biehn', 48),
('nm0000322', 'Emmanuelle Béart', 48),
('nm0435299', 'Meiko Kaji', 48),
('nm0124920', 'G. Larry Butler', 48),
('nm1649765', 'Samir Ghanem', 48),
('nm0597390', 'Kaori Momoi', 48),
('nm0814799', 'Jean Sorel', 48),
('nm0014109', 'Samira Ahmed', 48),
('nm0265252', 'Hussein Fahmy', 48),
('nm0000377', 'Richard Dreyfuss', 48),
('nm0000199', 'Al Pacino', 48),
('nm0000190', 'Matthew McConaughey', 48),
('nm7014801', 'Mieko Harada', 48),
('nm0001434', 'Kris Kristofferson', 48),
('nm0002181', 'Imanol Arias', 48),
('nm0000140', 'Michael Douglas', 48),
('nm0798328', 'Henry Silva', 48),
('nm0553445', 'Albert Martinez', 48),
('nm1004985', 'Yashpal Sharma', 48),
('nm0945999', 'Hui-Shan Yang', 48),

```
('nm0000515', 'Virginia Madsen', 48),  
( 'nm0128530', 'Engin Çağlar', 48),  
( 'nm0432005', 'Chun-Erh Lung', 48),  
( 'nm0034301', 'Rosita Arenas', 48),  
( 'nm0704133', 'Rosita Quintana', 48),  
( 'nm0001953', 'Moritz Bleibtreu', 48),  
( 'nm0271806', 'Polly Ling-Feng Shang-Kuan', 47),  
( 'nm0001287', 'Heather Graham', 47),  
( 'nm1333687', 'Dhanush', 47),  
( 'nm0155532', 'Adam Cheng', 47),  
( 'nm0559483', 'Yasuko Matsui', 47),  
( 'nm0000767', 'Jean-Hugues Anglade', 47),  
( 'nm0418440', 'Iva Janžurová', 47),  
( 'nm0000255', 'Ben Affleck', 47),  
( 'nm2754475', 'Rudranil Ghosh', 47),  
( 'nm0451383', 'Mukesh Khanna', 47),  
( 'nm1787828', 'Mayumi Inoue', 47),  
( 'nm0000093', 'Brad Pitt', 47),  
( 'nm0687914', 'Denis Podalydès', 47),  
( 'nm0156891', 'Kuan-Chun Chi', 47),  
( 'nm0768614', 'Leonardo Sbaraglia', 47),  
( 'nm0233425', 'Irma Dorantes', 47),  
( 'nm0490492', 'Tony Liu', 47),  
( 'nm0080149', 'Jaya Bachchan', 47),  
( 'nm0000249', 'James Woods', 47),  
( 'nm0000148', 'Harrison Ford', 47),  
( 'nm1045633', 'Liberty Ilagan', 47),  
( 'nm0002010', 'Christian Clavier', 47),  
( 'nm0688143', 'Benoît Poelvoorde', 47),  
( 'nm0000362', 'Danny DeVito', 47),  
( 'nm0704479', 'Eric Quizon', 47),  
...]
```

```
In [64]: Id_actor = StructField("nconst",StringType(),True)
Movie = StructField("Movie",StringType(),True)
nvotes = StructField("TotalActs",StringType(),True)

final_dataframe = sqlContext.createDataFrame(df_clean, StructType([Id_actor, M
ovie,nvotes])).persist()
```

```
+-----+-----+
      tconst|sum(TotalActs)|
+-----+-----+
tt3835486|          92.0|
tt7860370|          64.0|
tt0308851|          82.0|
tt5357670|          57.0|
tt3586950|          63.0|
tt3355560|           6.0|
tt2274007|          39.0|
tt2169873|           5.0|
tt10788536|           8.0|
tt3884798|          30.0|
tt3887522|          30.0|
tt4179840|          22.0|
tt7209900|          71.0|
tt7223904|          71.0|
tt7229340|          71.0|
tt7618978|          30.0|
tt7625502|          30.0|
tt2088382|           6.0|
tt6302248|          65.0|
tt0702398|           3.0|
+-----+-----+
only showing top 20 rows
```

```
In [65]: q11_MR = final_dataframe.join(title_principals,['nconst']).select(['nconst','T
otalActs','Movie','tconst']).distinct().groupBy('tconst').agg({'TotalActs':'su
m'})
q11_MR.show()
```



```
In [66]: #Q5
rank_renamed = page_rank.vertices.withColumnRenamed('id', 'nconst')

q11_PR = rank_renamed.join(title_principals, 'nconst').select(['tconst', 'nconst', 'pagerank']).distinct().groupBy('tconst').agg({'pagerank': 'avg'})
q11_PR.show()
```

```
+-----+-----+
      tconst|      avg(pagerank)|
+-----+-----+
tt3835486| 3.434041187049006|
tt7860370| 4.178459854543034|
tt0308851| 2.894842554684516|
tt5357670| 2.6987154184869393|
tt3586950| 3.0038843029997975|
tt3355560| 0.6448365075368346|
tt2274007| 1.3138991122555754|
tt2169873| 0.5810492369485097|
tt10788536| 0.633445120314059|
tt3884798| 1.2786099966876656|
tt3887522| 1.2786099966876656|
tt4179840| 1.1929589446056144|
tt7209900| 1.67554651434356|
tt7223904| 1.67554651434356|
tt7229340| 1.67554651434356|
tt7618978| 0.7470865498323299|
tt7625502| 0.7470865498323299|
tt2088382| 0.5868120721579081|
tt6302248| 2.4295941020504177|
tt0702398| 0.328289958489533|
+-----+-----+
only showing top 20 rows
```

```
In [67]: #Q6
df_out = ourGraph.outDegrees.withColumnRenamed("id", "nconst")

q11_OD = df_out.join(title_principals, 'nconst').select(['tconst', 'nconst', 'out
Degree']).distinct().groupBy('tconst').agg({'outDegree': 'avg'})
q11_OD.show()
```

```
+-----+-----+
      tconst|      avg(outDegree)|
+-----+-----+
tt3835486|          56.25|
tt7860370|          74.0|
tt0308851|          49.0|
tt5357670|          44.0|
tt3586950|          46.0|
tt3355560|           6.5|
tt2274007|13.833333333333334|
tt2169873|           1.5|
tt10788536|           4.0|
tt3884798|13.666666666666666|
tt3887522|13.666666666666666|
tt4179840|          11.0|
tt7209900|          19.75|
tt7223904|          19.75|
tt7229340|          19.75|
tt7618978|           7.2|
tt7625502|           7.2|
tt2088382|3.6666666666666665|
tt6302248|          21.5|
tt0702398|           2.0|
+-----+-----+
```

only showing top 20 rows

```
In [68]: #Q7
distance = distances_value.withColumnRenamed("id", "nconst").withColumnRenamed(
    "value", "distance")

q11_DD = distance.join(title_principals, 'nconst').select(['nconst', 'distance',
    'tconst']).distinct().groupBy('tconst').agg({'distance': 'avg'})
q11_DD.show()
```

```
+-----+-----+
      tconst|      avg(distance)|
+-----+-----+
tt3835486|                2.25|
tt7860370|                2.0|
tt0308851|                2.5|
tt5357670|2.6666666666666665|
tt3586950|                3.0|
tt3355560|                3.5|
tt2274007|                2.8|
tt2169873|                4.5|
tt10788536|                3.5|
tt3884798|3.6666666666666665|
tt3887522|3.6666666666666665|
tt4179840|                4.0|
tt7209900|                3.75|
tt7223904|                3.75|
tt7229340|                3.75|
tt7618978|                4.0|
tt7625502|                4.0|
tt2088382| 4.333333333333333|
tt6302248|                4.0|
tt0702398| 4.333333333333333|
+-----+-----+
only showing top 20 rows
```

```
In [69]: all_together = q11_MR.join(q11_PR, ['tconst'], 'inner').join(q11_OD, ['tconst'
    ], 'inner').join(q11_DD, ['tconst'], 'inner').join(title_ratings, ["tconst"], h
    ow="left_outer").drop('numVotes')
```

```
In [70]: # df_out = spark.createDataFrame(ourGraph.outDegrees, schema=['nconst', 'OutDegree'])
df_out = ourGraph.outDegrees.withColumnRenamed("id", "nconst")

q11_OD = df_out.join(title_principals, 'nconst').select(['tconst', 'nconst', 'outDegree']).distinct().groupBy('tconst').agg({'outDegree': 'avg'})
q11_OD.show()
```

```
+-----+-----+
   tconst|   avg(outDegree)|
+-----+-----+
tt3835486|          56.25|
tt7860370|          74.0|
tt0308851|          49.0|
tt5357670|          44.0|
tt3586950|          46.0|
tt3355560|           6.5|
tt2274007|13.833333333333334|
tt2169873|           1.5|
tt10788536|          4.0|
tt3884798|13.666666666666666|
tt3887522|13.666666666666666|
tt4179840|          11.0|
tt7209900|          19.75|
tt7223904|          19.75|
tt7229340|          19.75|
tt7618978|           7.2|
tt7625502|           7.2|
tt2088382|3.6666666666666665|
tt6302248|          21.5|
tt0702398|           2.0|
+-----+-----+
only showing top 20 rows
```

```
In [71]: all_together_without_nan = all_together.na.drop()
```

```
In [72]: #Don't need the tconst, numVotes and averageRating columns to create the features set
all_together_without_nan_without_tconst = all_together_without_nan.drop("tconst", "numVotes")
```

```
In [73]: # Import LinearRegression class
from pyspark.ml.regression import LinearRegression
from pyspark.ml.evaluation import RegressionEvaluator
from pyspark.sql import Row
from pyspark.ml.linalg import Vectors

train_test = all_together_without_nan_without_tconst.randomSplit([0.7,0.3], seed=0)
train = train_test[0]
test = train_test[1]
```

```
In [74]: from pyspark.ml.feature import VectorAssembler
features_Columns = all_together_without_nan_without_tconst.columns
features_Columns.remove("averageRating")
vectorAssembler = VectorAssembler(inputCols = features_Columns, outputCol = "features")
```

```
In [75]: from pyspark.ml import Pipeline
model = LinearRegression(labelCol = "averageRating")

pipeline = Pipeline(stages = [vectorAssembler, model])
```

```
In [76]: #Fitting the model
pipelineModel = pipeline.fit(train)
```

```
In [77]: pred = pipelineModel.transform(test)
print('My pre final answer is:')
print(pred.show())
```

My pre final answer is:

```
+-----+-----+-----+-----+-----+
-----+-----+
sum(TotalActs)|      avg(pagerank)|avg(outDegree)|avg(distance)|averageRating|
features|      prediction|
+-----+-----+-----+-----+-----+
-----+-----+
          1.0|0.14825619416216257|          1.0|          4.0|          7.1|[1.0,
0.1482561941...| 6.970018327882341|
          1.0|0.15544664939831743|          1.0|          3.0|          6.9|[1.0,
0.1554466493...| 7.190115028803963|
          1.0|0.15544664939831743|          1.0|          3.0|          7.2|[1.0,
0.1554466493...| 7.190115028803963|
          1.0| 0.1560848019826317|          1.0|          4.0|          7.0|[1.0,
0.1560848019...| 6.969011626921726|
          1.0|0.16356841718510448|          1.0|          4.0|          3.7|[1.0,
0.1635684171...| 6.968049289455605|
          1.0|0.19027504434168363|          1.0|          4.0|          7.6|[1.0,
0.1902750443...| 6.964615015082705|
          1.0|0.19781134273910975|          1.0|          5.0|          7.2|[1.0,
0.1978113427...| 6.742624562759856|
          1.0|0.20335914470855318|          1.0|          5.0|          5.9|[1.0,
0.2033591447...| 6.741911156532335|
          1.0|0.22067750796135932|          1.0|          5.0|          6.9|[1.0,
0.2206775079...| 6.739684143336285|
          1.0| 0.2234198063954968|          1.0|          4.0|          6.6|[1.0,
0.2234198063...| 6.960352844264581|
          1.0|0.24542602176007688|          2.0|          3.0|          8.1|[1.0,
0.2454260217...| 7.18028291658288|
          1.0| 0.2639889702916298|          2.0|          3.0|          8.1|[1.0,
0.2639889702...| 7.177895858941583|
          1.0|0.26718868500129506|          3.0|          3.0|          7.4|[1.0,
0.2671886850...| 7.179222967711467|
          1.0| 0.2724191599951741|          1.0|          4.0|          7.1|[1.0,
0.2724191599...| 6.954051890446285|
          1.0|0.27991829736925433|          1.0|          4.0|          9.1|[1.0,
0.2799182973...| 6.953087556943905|
          1.0| 0.289161640773263|          3.0|          3.0|          5.7|[1.0,
0.2891616407...|7.1763974084050774|
          1.0| 0.2991571700085228|          4.0|          3.0|          6.8|[1.0,
0.2991571700...| 7.176850625779713|
          1.0|0.29943561453647044|          3.0|          3.0|          6.6|[1.0,
0.2994356145...| 7.175076251502861|
          1.0|0.31236270176388836|          1.0|          4.0|          7.5|[1.0,
0.3123627017...| 6.948915446931539|
          1.0| 0.3230245125249654|          1.0|          4.0|          5.0|[1.0,
0.3230245125...| 6.947544417065027|
+-----+-----+-----+-----+-----+
-----+-----+
```

only showing top 20 rows

None

```
In [78]: from pyspark.ml.evaluation import RegressionEvaluator

RMSE = RegressionEvaluator(labelCol = "averageRating", predictionCol = "prediction", metricName = "rmse")
rmse = RMSE.evaluate(pred)

print('My final answer is: ' + str(rmse))

My final answer is: 1.335997160610406
```

Q12 What score would your model predict for the 1997 movie Titanic and how does this compare to it's actual score.

```
In [80]: df_filter.filter((df_filter['primaryTitle'].isin(['Titanic'])) & (df_filter["startYear"] == '1997')).select("tconst").show(1)

+-----+
| tconst |
+-----+
| tt0120338 |
+-----+
only showing top 1 row
```

```
In [81]: toPredict = all_together.filter(all_together.tconst == "tt0120338")
```

```
In [82]: #Preprocess to do the prediction
toPredict = pred.drop("tconst", "numVotes")
```

```
In [83]: toPredict.show(1)

+-----+-----+-----+-----+-----+-----+
| sum(TotalActs) | avg(pagerank) | avg(outDegree) | avg(distance) | averageRating |
| features | prediction |
+-----+-----+-----+-----+-----+-----+
| 1.0 | 0.14825619416216257 | 1.0 | 4.0 | 7.1 | [1.0, 0.1482561941... | 6.970018327882339 |
+-----+-----+-----+-----+-----+-----+
only showing top 1 row
```



```
In [84]: toPredict.select("averageRating", 'prediction').show()
```

```
+-----+-----+
averageRating|      prediction|
+-----+-----+
          7.1| 6.970018327882339|
          6.9| 7.1901150288039535|
          7.2| 7.1901150288039535|
          7.0| 6.9690116269217235|
          3.7| 6.968049289455603|
          7.6| 6.964615015082702|
          7.2| 6.742624562759861|
          5.9| 6.741911156532341|
          6.9| 6.739684143336291|
          6.6| 6.9603528442645795|
          8.1| 7.18028291658287|
          8.1| 7.177895858941573|
          7.4| 7.179222967711457|
          7.1| 6.954051890446284|
          9.1| 6.953087556943904|
          5.7| 7.176397408405068|
          6.8| 7.176850625779704|
          6.6| 7.175076251502852|
          7.5| 6.9489154469315375|
          5.0| 6.947544417065026|
+-----+-----+
only showing top 20 rows
```

Q13 Create dummy variables for each of the top 10 movie genres from **Q10**. These variable should have a value of 1 if the movie was rated with that genre and 0 otherwise. For example the 1997 movie Titanic should have a 1 in the dummy variable column for Romance, and a 1 in the dummy variable column for Drama, and 0's in all the other dummy variable columns.

If you were unable to answer Q10 you can just select 10 different genres and construct the same data.

Note: Question 10 uses the number of votes per genre and not the average votes per genre.

Does adding these variables to the regression improve your results? What is the new RMSE and predicted rating for the 1997 movie Titanic.

```
In [86]: rdd10_map.take(10)
```

```
Out[359]: [('Drama', 405996518),
 ('Action', 255651018),
 ('Comedy', 245067542),
 ('Adventure', 219323086),
 ('Crime', 149530637),
 ('Thriller', 138627057),
 ('Sci-Fi', 105588895),
 ('Romance', 103837192),
 ('Fantasy', 80317716),
 ('Mystery', 78414283)]
```

```
In [87]: df_filter.columns
```

```
Out[360]: ['tconst',
 'ordering',
 'nconst',
 'category',
 'job',
 'characters',
 'titleType',
 'primaryTitle',
 'originalTitle',
 'isAdult',
 'startYear',
 'endYear',
 'runtimeMinutes',
 'genres',
 'primaryName',
 'birthYear',
 'deathYear',
 'primaryProfession',
 'knownForTitles']
```

```
In [88]: # from pyspark.sql import functions as F
```

```
# df = sqlContext.createDataFrame([
#     (1, "a"),
#     (2, "b"),
#     (3, "c"),
# ], ["ID", "Text"])

# categories = df.select("Text").distinct().rdd.flatMap(lambda x: x).collect()

# exprs = [F.when(F.col("Text") == category, 1).otherwise(0).alias(category)
#           for category in categories]

# df.select("ID", *exprs).show()

movie_genre = df_filter.rdd.map(lambda x: (x[13], x[0]))
```

```
In [89]: movie_genre.take(1)
```

```
Out[362]: [('Biography,Drama,Music', 'tt0110116')]
```

```
In [90]: movie_genre = movie_genre.map(lambda x: (x[0].split(','),x[1])).map(lambda x: [(y, x[1]) for y in x[0]]).flatMap(lambda x: x)
```

```
In [91]: movie_genre.take(1)
```

```
Out[364]: [('Biography', 'tt0110116')]
```

```
In [92]: from pyspark.sql.types import *
from pyspark.sql.types import StructField
from pyspark.sql.functions import regexp_replace,col

movie_genre = movie_genre.map(lambda x: (x[1],x[0]))

tconts = StructField("ID",StringType(), True)
genre = StructField("genre",StringType(), True)

dummy = sqlContext.createDataFrame(movie_genre, StructType([tconts, genre]))
genre = rdd10_map.map(lambda x: x[0]).take(10)
```

```
In [93]: dummy = dummy.filter(dummy.genre.isin(genre))
```

```
In [94]: dummy.show(1)
```

```
+-----+-----+
      ID|genre|
+-----+-----+
tt0110116|Drama|
+-----+-----+
only showing top 1 row
```

```
In [95]: from pyspark.sql import functions as func
dummy = dummy.groupBy("ID").pivot("genre").agg(func.lit(1)).na.fill(0)
```

In [96]: `dummy.columns`

```
Out[370]: ['ID',
           'Action',
           'Adventure',
           'Comedy',
           'Crime',
           'Drama',
           'Fantasy',
           'Mystery',
           'Romance',
           'Sci-Fi',
           'Thriller']
```

In [97]: `dummy.show(5)`

```
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
---+
      ID|Action|Adventure|Comedy|Crime|Drama|Fantasy|Mystery|Romance|Sci-Fi|Thrill
er|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
---+
tt0089167|    0|        0|    1|    0|    0|    0|    0|    0|    0|
0|
tt0098048|    1|        0|    0|    0|    0|    1|    0|    0|    1|
0|
tt8400856|    0|        0|    1|    0|    1|    0|    0|    0|    0|
0|
tt3055374|    0|        0|    0|    0|    0|    0|    0|    0|    0|
1|
tt0074792|    0|        0|    0|    1|    1|    0|    0|    0|    0|
1|
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
---+
only showing top 5 rows
```

In [98]: `Final_dummy = all_together_without_nan.join(dummy, all_together_without_nan.tc
onst == dummy.ID, how = 'left').na.fill(0)`

In [99]: `Final_dummy.show(1)`

```

In [100]: #With the new dummy model
train_test = Final_dummy.randomSplit([0.7,0.3], seed=0)
train = train_test[0]
test = train_test[1]

from pyspark.ml.feature import VectorAssembler
features_Columns = Final_dummy.columns
features_Columns.remove("averageRating")
vectorAssembler = VectorAssembler(inputCols = features_Columns, outputCol = "features")

from pyspark.ml import Pipeline
model = LinearRegression(labelCol = "averageRating")

pipeline = Pipeline(stages = [vectorAssembler, model])
#Fitting the model
pipelineModel = pipeline.fit(train)
pred = pipelineModel.transform(test)
print('My pre final answer is:')
print(pred.show())

```

Q14 Improve your model by testing different machine learning algorithms, using hyperparameter tuning on these algorithms, changing the included features. Be careful not to cheat and use test data in the training of your model.

Note: We are not testing your knowledge of different algorithms, we are just testing that you can apply the different tools in the spark toolkit and can compare between them.

What is the RMSE of you final model and what rating does it predict for the 1997 movie Titanic.

```

In [102]: from pyspark.ml.feature import VectorAssembler
features_Columns = all_together_without_nan_without_tconst.columns
features_Columns.remove("averageRating")
vectorAssembler = VectorAssembler(inputCols = features_Columns, outputCol = "features")

```

```

In [103]: from pyspark.ml.regression import GBRegressor # GB
from pyspark.ml.regression import RandomForestRegressor # RF

train_test = all_together_without_nan_without_tconst.randomSplit([0.7,0.3], seed=0)
train = train_test[0]
test = train_test[1]

from pyspark.ml import Pipeline
GBModel = GBRegressor(labelCol = "averageRating")

```

```
In [104]: from pyspark.ml.tuning import CrossValidator, ParamGridBuilder
paramGrid = ParamGridBuilder().addGrid(GBTmodel.maxDepth, [4,8]).build()

evaluator = RegressionEvaluator(metricName="rmse", labelCol=GBTmodel.getLabelCol(), predictionCol=GBTmodel.getPredictionCol())
cv = CrossValidator(estimator = GBTmodel, evaluator = evaluator, estimatorParamMaps=paramGrid)
```

```
In [105]: pipeline = Pipeline(stages=[vectorAssembler, vectorIndexer, cv])
```

```
In [106]: pipelineModel = pipeline.fit(train)
pred = pipelineModel.transform(test)
print('My pre final answer is:')
print(pred.show())
```