Examen_SD_VOP

June 27, 2019

1. Adquirir los datos. Se descargó el archivo mediante el comando: !wget https://archive.org/download/nycTaxiTripData2013/trip_data.7z Se tardó 9:22 min. -> 3.82G 10.7MB/s

2. Una vez descargado el archivo pueden descomprimirlo usando la siguiente instrucción

In [ ]:
```
!wget https://archive.org/download/nycTaxiTripData2013/trip_data.7z
!7z x trip_data.7z -o/content/
```

In [1]:
```
!ls -lh
```

```
total 27G
-rw-r--r-- 1 kl kl 124K jun 26 22:47 Examen_SD_VOP.ipynb
-rw-r--r-- 1 kl kl 396K jun 26 15:08 Examen_SistemasDistribuidos_Dask.ipynb
-rw-r--r-- 1 kl kl 743K jun 26 22:05 Examen_SistemasDistribuidos_Dask_vf.ipynb
-rwxr-xr-x 1 kl kl 2.4G ene 15  2014 trip_data_10.csv
-rwxr-xr-x 1 kl kl 2.3G ene 15  2014 trip_data_11.csv
-rwxr-xr-x 1 kl kl 2.2G ene 15  2014 trip_data_12.csv
-rwxr-xr-x 1 kl kl 2.3G may 12  2014 trip_data_1.csv
-rwxr-xr-x 1 kl kl 2.2G may 12  2014 trip_data_2.csv
-rwxr-xr-x 1 kl kl 2.5G ago 25  2013 trip_data_3.csv
-rwxr-xr-x 1 kl kl 2.4G ago 25  2013 trip_data_4.csv
-rwxr-xr-x 1 kl kl 2.4G ago 25  2013 trip_data_5.csv
-rwxr-xr-x 1 kl kl 2.3G ago 25  2013 trip_data_6.csv
-rwxr-xr-x 1 kl kl 2.2G ene 14  2014 trip_data_7.csv
-rwxr-xr-x 1 kl kl 2.0G ene 14  2014 trip_data_8.csv
-rwxr-xr-x 1 kl kl 2.2G ene 15  2014 trip_data_9.csv
-rw-r--r-- 1 kl kl  44K jun 26 21:04 Untitled.ipynb
```

In [2]:
```
import numpy as np
import pandas as pd
import psutil, os

def huella_memoria():
    '''Regresa la huella de memoria en MB usada por un proceso de Python'''
    mem = psutil.Process(os.getpid()).memory_info().rss
    return (mem / 1024**2)
```

```
antes = huella_memoria()
%time df = pd.read_csv('trip_data_1.csv')
despues = huella_memoria()
print ('Huella de memoria: {} MB'.format(despues-antes))
print ('Memoria del sistema: {} MB'.format(sum(df.memory_usage(index=False) /1024**2)))
```

```
CPU times: user 55.5 s, sys: 3.79 s, total: 59.3 s
Wall time: 1min 4s
Huella de memoria: 3225.76171875 MB
Memoria del sistema: 1578.312759399414 MB
```

```
In [3]: df.shape
```

```
Out[3]: (14776615, 14)
```

```
In [4]: df.columns
```

```
Out[4]: Index(['medallion', 'hack_license', 'vendor_id', 'rate_code',
               'store_and_fwd_flag', 'pickup_datetime', 'dropoff_datetime',
               'passenger_count', 'trip_time_in_secs', 'trip_distance',
               'pickup_longitude', 'pickup_latitude', 'dropoff_longitude',
               'dropoff_latitude'],
              dtype='object')
```

3. Responder las siguientes preguntas:

    1. Cuantas columnas contiene cada archivo de datos descomprimido.
    2. Cuantos renglones tiene cada archivo.

Después de cargar el archivo con la librería pandas se utiliza la propiedad "shape" para obtener que el archivo cuenta con 14 columnas y 14,776,615 renglones. Las columnas son: medallion hack_license vendor_id rate_code store_and_fwd_flag pickup_datetime dropoff_datetime passenger_count trip_time_in_secs trip_distance pickup_longitude pickup_latitude dropoff_longitude dropoff_latitude

```
In [5]: df.head()
```

```
Out[5]:                           medallion                      hack_license  \
        0  89D227B655E5C82AECF13C3F540D4CF4  BA96DE419E711691B9445D6A6307C170
        1  0BD7C8F5BA12B88E0B67BED28BEA73D8  9FD8F69F0804BDB5549F40E9DA1BE472
        2  0BD7C8F5BA12B88E0B67BED28BEA73D8  9FD8F69F0804BDB5549F40E9DA1BE472
        3  DFD2202EE08F7A8DC9A57B02ACB81FE2  51EE87E3205C985EF8431D850C786310
        4  DFD2202EE08F7A8DC9A57B02ACB81FE2  51EE87E3205C985EF8431D850C786310

          vendor_id  rate_code store_and_fwd_flag      pickup_datetime  \
        0       CMT          1                  N  2013-01-01 15:11:48
        1       CMT          1                  N  2013-01-06 00:18:35
        2       CMT          1                  N  2013-01-05 18:49:41
        3       CMT          1                  N  2013-01-07 23:54:15
```

```
4       CMT         1                  N  2013-01-07 23:25:03

      dropoff_datetime  passenger_count  trip_time_in_secs  trip_distance  \
0  2013-01-01 15:18:10                4                382            1.0
1  2013-01-06 00:22:54                1                259            1.5
2  2013-01-05 18:54:23                1                282            1.1
3  2013-01-07 23:58:20                2                244            0.7
4  2013-01-07 23:34:24                1                560            2.1

   pickup_longitude  pickup_latitude  dropoff_longitude  dropoff_latitude
0        -73.978165        40.757977         -73.989838         40.751171
1        -74.006683        40.731781         -73.994499         40.750660
2        -74.004707        40.737770         -74.009834         40.726002
3        -73.974602        40.759945         -73.984734         40.759388
4        -73.976250        40.748528         -74.002586         40.747868
```

In [6]:
```
df.tail()
```

Out[6]:
```
                                 medallion                      hack_license  \
14776610  B33E71CD9E8FE1BE3B70FEB6E807DD15  BAF57796E45D921BB23217E17A372FF6
14776611  ED160B76D5349C8AC1ECF22CD4B8D538  3B93F6DA5DEBDE9560993FA624C4FF76
14776612  D83F9AC0E33F6F19869C243BE6AB6FE5  85A55B6772275374EF90AC9457DC1F83
14776613  04E59442A7DDBCE515E33CD355D866E7  7913172189931A1A1632562B10AB53C4
14776614  D30BED60331C79E3F7ACD05B325ED42F  B5E1D2461A5BCC8819188DACEC17CD69

         vendor_id  rate_code store_and_fwd_flag      pickup_datetime  \
14776610       CMT          1                  N  2013-01-06 04:58:23
14776611       CMT          1                  N  2013-01-08 14:42:04
14776612       CMT          1                  N  2013-01-10 13:29:23
14776613       CMT          1                  N  2013-01-06 16:30:15
14776614       CMT          1                  N  2013-01-05 20:38:46

             dropoff_datetime  passenger_count  trip_time_in_secs  \
14776610  2013-01-06 05:11:24                1                781
14776611  2013-01-08 14:50:27                1                503
14776612  2013-01-10 13:34:45                1                321
14776613  2013-01-06 16:42:26                1                730
14776614  2013-01-05 20:43:06                1                260

          trip_distance  pickup_longitude  pickup_latitude  dropoff_longitude  \
14776610            3.3        -73.989029        40.759327         -73.953743
14776611            1.0        -73.993042        40.733990         -73.982483
14776612            0.9        -73.979553        40.785011         -73.968262
14776613            1.3        -73.968002        40.762161         -73.985992
14776614            0.8        -73.982224        40.766670         -73.989212

          dropoff_latitude
14776610         40.770672
```

```
14776611        40.724823
14776612        40.788158
14776613        40.770542
14776614        40.773636
```

```
In [7]: # Verificando registros sin Medallón
        df[df['medallion'].str.len()<30]
```

```
Out[7]: Empty DataFrame
        Columns: [medallion, hack_license, vendor_id, rate_code, store_and_fwd_flag, pickup_da
        Index: []
```

```
In [8]: # Verificando registros sin licencia
        df[df['hack_license'].str.len()<30]
```

```
Out[8]: Empty DataFrame
        Columns: [medallion, hack_license, vendor_id, rate_code, store_and_fwd_flag, pickup_da
        Index: []
```

```
In [9]: df['vendor_id'].unique()
```

```
Out[9]: array(['CMT', 'VTS'], dtype=object)
```

```
In [10]: # Verificando registros sin vendedor
         df[df['vendor_id'].str.len()<3]
```

```
Out[10]: Empty DataFrame
         Columns: [medallion, hack_license, vendor_id, rate_code, store_and_fwd_flag, pickup_d
         Index: []
```

```
In [11]: df['rate_code'].unique()
```

```
Out[11]: array([  1,   2,   4,   5,   3,   6,   8,   0, 210,  28,   7,   9,  65,
                128])
```

```
In [12]: df[df['rate_code']>9]
```

```
Out[12]:                                  medallion                      hack_license  \
         2540625   F5818A7D75529449112CD423EA58626C  3854656E0C4C9BA60A26295D68EF4EE8
         2982293   61C22E72D833C1F868293A9DE377CA8E  0FC602C2A935F5061F1012DBE9A893E5
         3389092   85666EE5D2EDFDAC22041D2B6ACF0D71  20558EC4435688C67A76F05C1CB216D8
         4055114   E523F84E85708BCEB9FEB6F7825C0E08  8B9EB8B7A837B4C28A2C54D3AEB6A442
         6255682   7EC413823306154C3808A882DF771A1E  A070F43524E97B976AA05260823123D6
         6759185   C499715B8AC7FC95E2E24C56BDB957C9  F9B944F6E93AC9F699488AC14DB6245B
         6841339   A38C0393DB4A916A99E4B345B2F70D92  E130CCC77720F124DE3D10C498700CC7
         6867270   4BA19608138BFFCCAEC0DEB910D5BFBE  957178F85806D8B2EA6BBAC8D65A368A
         7035307   E523F84E85708BCEB9FEB6F7825C0E08  305F98E26E9C7EDA65EF92131C01198C
         7096961   F6B90945F15CDB501B0C18E2163BF96E  04D102ACF5ED6B96174ED451ADB60EB7
         7958549   8C4F18BF18A8ECAB898369409A09CC17  3FA92EF60AD7FFA87266AD35B76B1BC4
         10724973  11FB95AC57D4AA4CB82BB2E061C53374  75950CC7D3A612585E645509BE9636C8
```

11193753	E523F84E85708BCEB9FEB6F7825C0E08	305F98E26E9C7EDA65EF92131C01198C
11777985	EFB08EF1CDA77CC654234F098E06E90C	A91625A6004F32A7482A209D5A1BC174
12513429	41D2AC8217D65CF5B493D009BEC5221E	0AC2063528084286DE7626476DD645F8
12518428	41D2AC8217D65CF5B493D009BEC5221E	0AC2063528084286DE7626476DD645F8
12518752	41D2AC8217D65CF5B493D009BEC5221E	0AC2063528084286DE7626476DD645F8
12642203	41D2AC8217D65CF5B493D009BEC5221E	0AC2063528084286DE7626476DD645F8

	vendor_id	rate_code	store_and_fwd_flag	pickup_datetime	\
2540625	CMT	210	N	2013-01-02 20:07:05	
2982293	CMT	210	N	2013-01-06 07:11:17	
3389092	CMT	210	N	2013-01-10 11:23:27	
4055114	CMT	28	N	2013-01-15 17:30:34	
6255682	CMT	210	N	2013-01-03 16:17:51	
6759185	CMT	210	N	2013-01-15 11:36:46	
6841339	CMT	210	N	2013-01-02 18:01:00	
6867270	CMT	210	N	2013-01-15 11:30:37	
7035307	CMT	28	N	2013-01-14 01:32:49	
7096961	CMT	210	N	2013-01-10 13:56:57	
7958549	CMT	210	N	2013-01-21 18:28:30	
10724973	CMT	210	N	2013-01-31 15:33:15	
11193753	CMT	65	N	2013-01-28 01:18:22	
11777985	CMT	210	N	2013-01-19 13:23:20	
12513429	VTS	128	NaN	2013-01-09 10:27:00	
12518428	VTS	128	NaN	2013-01-09 10:19:00	
12518752	VTS	128	NaN	2013-01-09 10:23:00	
12642203	VTS	128	NaN	2013-01-09 19:34:00	

	dropoff_datetime	passenger_count	trip_time_in_secs	\
2540625	2013-01-02 20:20:59	1	833	
2982293	2013-01-06 07:20:12	1	535	
3389092	2013-01-10 11:33:55	1	628	
4055114	2013-01-15 18:52:00	1	4886	
6255682	2013-01-03 16:30:43	1	771	
6759185	2013-01-15 11:49:59	1	793	
6841339	2013-01-02 18:19:30	4	1109	
6867270	2013-01-15 11:50:29	3	1191	
7035307	2013-01-14 02:11:12	1	2303	
7096961	2013-01-10 14:36:06	1	2349	
7958549	2013-01-21 18:44:08	1	938	
10724973	2013-01-31 16:02:27	3	1752	
11193753	2013-01-28 02:09:36	1	3074	
11777985	2013-01-19 13:44:51	2	1290	
12513429	2013-01-09 10:28:00	1	60	
12518428	2013-01-09 10:21:00	1	120	
12518752	2013-01-09 10:24:00	1	60	
12642203	2013-01-09 20:06:00	1	1920	

	trip_distance	pickup_longitude	pickup_latitude	dropoff_longitude	\

```
2540625           2.6        -73.987122        40.752220        -73.988487
2982293           5.3        -73.954391        40.789707        -73.944160
3389092           1.7        -73.979828        40.765343        -73.976311
4055114          13.4        -73.843529        40.733173        -73.906349
6255682           1.2        -73.991890        40.744198        -73.997925
6759185           2.2        -74.005051        40.719086        -73.986336
6841339           1.5        -73.973732        40.763069        -73.992729
6867270           5.2        -74.005043        40.707088        -73.999611
7035307           2.2        -73.994225        40.755878        -73.951408
7096961          19.2        -73.955849        40.778938        -73.782364
7958549           4.6        -73.975777        40.749882        -73.943130
10724973          5.5        -73.989326        40.773426        -73.987228
11193753          5.5        -73.987167        40.729111        -73.950867
11777985          3.6        -73.970078        40.759342        -74.004150
12513429          0.0          0.000000         0.000000          0.000000
12518428          0.0          0.000000         0.000000          0.000000
12518752          0.0          0.000000         0.000000          0.000000
12642203          0.0          0.000000         0.000000          0.000000

          dropoff_latitude
2540625          40.719772
2982293          40.843048
3389092          40.744682
4055114          40.745289
6255682          40.756237
6759185          40.739906
6841339          40.756718
6867270          40.763000
7035307          40.825195
7096961          40.644165
7958549          40.768764
10724973         40.715248
11193753         40.714134
11777985         40.747898
12513429          0.000000
12518428          0.000000
12518752          0.000000
12642203          0.000000
```

In [13]:
```
df['store_and_fwd_flag'].unique()
```

Out[13]:
```
array(['N', 'Y', nan], dtype=object)
```

In [14]:
```
df[df['pickup_datetime'].str.len()<15]
```

Out[14]:
```
Empty DataFrame
Columns: [medallion, hack_license, vendor_id, rate_code, store_and_fwd_flag, pickup_da
Index: []
```

In [15]:
```
df[df['dropoff_datetime'].str.len()<15]
```

Out[15]:
```
Empty DataFrame
Columns: [medallion, hack_license, vendor_id, rate_code, store_and_fwd_flag, pickup_da
Index: []
```

In [16]:
```
df['passenger_count'].unique()
```

Out[16]:
```
array([  4,   1,   2,   3,   5,   6,   0, 208,   9, 255])
```

In [17]:
```
df[df['passenger_count']>9]
```

Out[17]:
```
                                 medallion                      hack_license  \
1261636   ECCBC87E4B5CE2FE28308FD9F2A7BAF3  08B595B9B70457A1838B11B75C730EB8
12308722  DCDD025C953CA1BAAA7044C69BCE2D3D  3C1BBDEA84C6BF2B166483B676D362C1

         vendor_id  rate_code store_and_fwd_flag      pickup_datetime  \
1261636        VTS          1                NaN  2013-01-02 15:11:00
12308722       VTS          1                NaN  2013-01-08 10:16:00

             dropoff_datetime  passenger_count  trip_time_in_secs  \
1261636   2013-01-02 15:12:00              208                 60
12308722  2013-01-08 10:24:00              255                480

          trip_distance  pickup_longitude  pickup_latitude  dropoff_longitude  \
1261636             0.0       -115.231020        36.218132         -115.23106
12308722            1.7        -73.946877        40.714642         1347.44460

          dropoff_latitude
1261636          36.218121
12308722        898.293820
```

In [18]:
```
df['trip_time_in_secs'].unique()
```

Out[18]:
```
array([ 382,  259,  282, ..., 7034, 7749, 5471])
```

In [19]:
```
df[df['trip_time_in_secs']>9000]
```

Out[19]:
```
                                 medallion                      hack_license  \
29526     C6C2AA4F89E3C1ED8B58FFC7AE6D931C  F49FD0D84449AE7F72F3BC492CD6C754
113545    DBD60AF056CB92BD0C5C6EC6DB30A9FD  3DEE79430FC260A7D2276CA9CA46021E
157147    B28C65FAE2D2ADEF21FC1279B022E176  A6519EA2BD56AFB2BE217E085A8310C2
256114    4B9BC2FD131B7D4F09AD91D1EAE4D8CF  DDC26C1F1D9192548FB278F14931A471
486795    B91CBA168CDFF29638EF93E7F4D50E55  C488C7AF97BF082CD1DF065A3C163C3B
900367    13EAB123CDFA67EFCEB31388D17B2A2C  4489A28CD0680AB62D07701A0E6C2AFD
933085    E438441E8EEDB4CA6FDF7D6639DB1672  783832AFA60404D80FA92C4CFA379FF9
958662    5AC6992C1585FD1821B26E4952EF1F46  29473B668DD15779D7158C176363D191
958809    8FA70F4D885EDC95AE1984BF965C783B  AD21812F01BB6E84F542709B7B166A33
1196135   4B9BC2FD131B7D4F09AD91D1EAE4D8CF  DDC26C1F1D9192548FB278F14931A471
```

1211812	ABA35D2B038DCFBE042FB77F6DC49C88	AFC52A2738D63F9C2230A6DC19EC7B5F
1310253	13EAB123CDFA67EFCEB31388D17B2A2C	4489A28CD0680AB62D07701A0E6C2AFD
1570703	C96BFFF929E6ED33760A63F5B779C0C6	CCAAB67C9ACE343C21DEFC58BFED28F1
1720244	69AAA9E6753311E71926AA25F92E4820	94C91DBE848CF2D688611013958218CD
1844957	88F2AC0FDDF9BB5310661A5238179295	14BC717F3C0D72F92C4CB0C9DE6443EE
1879615	DBD60AF056CB92BD0C5C6EC6DB30A9FD	3DEE79430FC260A7D2276CA9CA46021E
1894145	77D9307463817B2441A5FDCBC87C0569	3A6D87A3F9E5826A663D13C1DB5534EA
1912151	233AF667F8819B792F8F8511A194EFDE	318651E071D292B217ED1635719C7E90
2149604	6F6DB4FE34210F8B7B47C01B136FFFC1	B16455EFCBBE46AB2B1BB7463F834FD7
2481328	F68264CC7BB246243FAC08F878579C2C	9513EE9A3828687C35908A89BDA654E3
2514809	69D85F0E194FF6503210607BCE671309	3DA88E56C2B90EB7B3D20F4C4719D869
2523595	18D226BF7FFEE1775B8117A9704191C2	98A59A3563801A8D3670AFFAF6DE4729
2541881	7F344E842AAC8F7FBC4AEA1C53CD209E	6385ABB50EAADE59BE55042694D2AE6A
2573058	7CAEB61CF1E248640B162BFE83F6C6A4	6499D0731862696945485AA5E37727C3
2574328	C560736682D2F1C6DB526F939AA01008	EA1A911B1599B2BBC0FC085AD8B113F6
2592899	8B3CEF0A600F56443E4CD2D7ADDDC860	7541E66430A5AEA6F211D0AD9B776DCD
2608234	E53A4C7E42EEF65BEF49F0E59C598489	6ACA96741757AD0B9E599610ED1FB4B5
2653124	FF2DE23F93FEC7FB11E82913FA575F94	BA4527CF0724B3C024C9132265B6D07B
2785138	970160B87955704A5FD6BD6817936BB4	0FB190339E7BA5797762A8252B6177F6
2785195	473E20C6CBE91A07C2CA161158E35AE4	7383602B95C4EA2BBCA77B21E43D06D1
...	...	...
12330035	4D6497519929FAAD503568D18CD21B30	4957D08150E238A69CF6D36CC8734A07
12445211	26C5365B5F8A4495DA0A261F926C53DB	0A08EC79EF0E7F3C14DC4AD92D27F97F
12541148	0220580F4DB64D1753A839B91840BFC9	00B7691D86D96AEBD21DD9E138F90840
12612492	E21D302C8F9C748BC3AFCEDE461EE2A5	56EDFDE70D288DE83CDF4BA311F9986F
12626775	53CFF2D43A674B63FFD75750753D9051	3AB8E3D4009CA74DFC3988918B45D62A
12664669	38E50730900AE2F4F2ABE518CCBBED70	3E6233DC8EA366096653DE71E8C3A27A
13063516	A6D0858338A830AEEF332B8540B497F3	F1FFF823D04F67D9B3B7515675EF78AB
13160480	0495CB05512673F9AE263421650F938E	9112D33A328C37CF6E8A6B364F0C6109
13184498	4CF38C0C5DE00C68B5AFFF44857FE359	10C8269A33156E44EA0237CA17F10916
13237102	EEF3A194E04ECF7B2CC01D7719179040	15DB9F70261C9AB6155213E56D06E6A6
13367219	C1A9F4FEC00376BA54B6CF5485AB3838	BDDD6763526EA25A74D8927929A4ED28
13540502	2AC118343392C57CF6FB7CE4EFA6C4D1	FE6397A4E1882B737821E2EDE3364D5F
13544928	2D33D342F8EC93B7E6B15ABCB05BD2CB	2295E5AA38EBF125E395EBE5DF20C35D
13546371	429AD5759D9F317802F3A6C5F138E7B9	8642DE0F37F1446515FBB7767C808548
13653505	1613484252A0D15B4AD2724B3A32D3CF	56E8D2297EF8373C45498FD34F0981F0
13655713	FC6FEC823E24A0D75CF5494A4FDA5200	685067519D9A05E1A2E2521A912BE15A
13684403	B2CEC8C5B4105A929E421200F7426D59	9EAF1B30EB71ABF7B377CC76E99BCB59
14151908	429AD5759D9F317802F3A6C5F138E7B9	2FA44E9C0C08DBA37B6B18F09685205D
14171902	7643C20C8B67F8905AB6D8CF3D705D92	A11C9B542540C823D67D85AE461FA674
14216991	7D12FC75A27AE3ECD0010D0A3987AC96	3DCFFDCAD260B320B1BDC7958DEFB428
14241138	8932913309067858D4152CF4771779BA	EAA7B4D5AC25D40CB4BEACE59582B60B
14351954	AC7E1AA476492D6006F29C7B9993D1DF	5F2186EFA349D0906D68E4E62F01790A
14373172	A94CC7B3B0A79E74DF9CAAF58511CB70	0CE4197F332E22E70122D6A325182953
14409370	1C22897F58EB9F97C33B441844E834AF	BD9342C1ACB114EC36181FC47E2977A7
14549307	55FA5A677288ABB9EC6ECA558E045E6D	AEED06149422C2ABC0D4C17E4F381860
14669733	FBDEAE151E908F556EE0E123ADF5E484	C742CFD86A6B2ABFB9CD7228286766CA
14669916	FBDEAE151E908F556EE0E123ADF5E484	C742CFD86A6B2ABFB9CD7228286766CA

14695019	4A062EFDCA34C0785EB7E60BEEFAD433	612775933EDE80BBC37394648AD22B1E
14739586	A3F2562998085F1B8FBAAD9BDCFE4034	BB84C8FDE2ACEBEBF2733D0A8D3A00FE
14751945	49ED42528DD7181E2DDE2F96C7EE4F16	3AE96A60FAE8215799C847F03EA63198

	vendor_id	rate_code	store_and_fwd_flag	pickup_datetime \
29526	VTS	1	NaN	2013-01-13 05:20:00
113545	VTS	2	NaN	2013-01-13 13:10:00
157147	VTS	1	NaN	2013-01-14 01:56:00
256114	VTS	2	NaN	2013-01-14 13:12:00
486795	VTS	2	NaN	2013-01-15 13:10:00
900367	VTS	1	NaN	2013-01-17 02:13:00
933085	VTS	1	NaN	2013-01-01 01:03:00
958662	VTS	1	NaN	2013-01-01 01:14:00
958809	VTS	2	NaN	2013-01-01 01:53:00
1196135	VTS	2	NaN	2013-01-02 12:28:00
1211812	VTS	1	NaN	2013-01-02 12:57:00
1310253	VTS	1	NaN	2013-01-03 03:37:00
1570703	VTS	2	NaN	2013-01-04 06:11:00
1720244	VTS	5	NaN	2013-01-04 18:41:00
1844957	VTS	2	NaN	2013-01-05 09:35:00
1879615	VTS	2	NaN	2013-01-05 13:46:00
1894145	VTS	2	NaN	2013-01-05 13:04:00
1912151	VTS	2	NaN	2013-01-05 15:18:00
2149604	VTS	2	NaN	2013-01-06 15:21:00
2481328	CMT	1	N	2013-01-01 20:19:45
2514809	CMT	2	Y	2013-01-02 06:06:12
2523595	CMT	1	N	2013-01-02 06:47:51
2541881	CMT	1	N	2013-01-02 14:45:07
2573058	CMT	1	N	2013-01-03 05:57:49
2574328	CMT	2	N	2013-01-03 05:47:12
2592899	CMT	1	N	2013-01-02 23:17:14
2608234	CMT	1	N	2013-01-03 11:24:13
2653124	CMT	1	N	2013-01-03 23:54:39
2785138	CMT	5	N	2013-01-05 08:22:59
2785195	CMT	5	N	2013-01-05 08:35:04
...	...	...	...	...
12330035	VTS	1	NaN	2013-01-08 09:31:00
12445211	VTS	1	NaN	2013-01-08 18:47:00
12541148	VTS	1	NaN	2013-01-09 11:06:00
12612492	VTS	1	NaN	2013-01-09 15:31:00
12626775	VTS	1	NaN	2013-01-09 12:44:00
12664669	VTS	1	NaN	2013-01-09 17:00:00
13063516	VTS	1	NaN	2013-01-11 13:10:00
13160480	VTS	2	NaN	2013-01-11 19:53:00
13184498	CMT	1	N	2013-01-02 16:27:39
13237102	VTS	2	NaN	2013-01-04 15:36:00
13367219	VTS	2	NaN	2013-01-12 15:51:00
13540502	CMT	1	N	2013-01-10 10:13:27

13544928	CMT	1	N	2013-01-02 01:58:44
13546371	CMT	1	Y	2013-01-02 00:09:12
13653505	CMT	1	N	2013-01-08 19:42:36
13655713	CMT	2	N	2013-01-11 16:52:25
13684403	CMT	1	N	2013-01-24 10:19:26
14151908	CMT	2	N	2013-01-27 12:09:05
14171902	CMT	2	N	2013-01-27 16:07:04
14216991	CMT	1	N	2013-01-23 07:37:45
14241138	CMT	5	N	2013-01-21 17:24:00
14351954	CMT	1	N	2013-01-21 16:57:11
14373172	CMT	1	N	2013-01-22 07:35:16
14409370	CMT	1	Y	2013-01-23 17:43:50
14549307	CMT	1	N	2013-01-26 09:46:59
14669733	CMT	1	N	2013-01-06 16:02:53
14669916	CMT	1	N	2013-01-06 21:10:21
14695019	CMT	1	N	2013-01-07 08:47:08
14739586	CMT	4	N	2013-01-01 12:57:54
14751945	CMT	1	N	2013-01-08 12:27:58

	dropoff_datetime	passenger_count	trip_time_in_secs	\
29526	2013-01-13 07:53:00	4	9180	
113545	2013-01-13 15:47:00	6	9420	
157147	2013-01-14 04:28:00	5	9120	
256114	2013-01-14 16:01:00	1	10140	
486795	2013-01-15 15:50:00	2	9600	
900367	2013-01-17 05:05:00	1	10320	
933085	2013-01-01 03:39:00	6	9360	
958662	2013-01-01 04:04:00	1	10200	
958809	2013-01-01 04:53:00	1	10800	
1196135	2013-01-02 15:14:00	2	9960	
1211812	2013-01-02 15:55:00	5	10680	
1310253	2013-01-03 06:35:00	1	10680	
1570703	2013-01-04 08:56:00	1	9900	
1720244	2013-01-04 21:15:00	1	9240	
1844957	2013-01-05 12:34:00	1	10740	
1879615	2013-01-05 16:17:00	6	9060	
1894145	2013-01-05 16:01:00	1	10620	
1912151	2013-01-05 18:09:00	1	10260	
2149604	2013-01-06 18:03:00	1	9720	
2481328	2013-01-01 23:00:55	1	9670	
2514809	2013-01-02 08:44:41	1	9509	
2523595	2013-01-02 09:22:17	1	9266	
2541881	2013-01-02 17:25:56	1	9649	
2573058	2013-01-03 08:36:15	1	9506	
2574328	2013-01-03 08:33:17	2	9965	
2592899	2013-01-03 01:55:22	1	9488	
2608234	2013-01-03 14:01:36	1	9443	
2653124	2013-01-04 02:29:18	1	9279	

2785138	2013-01-05 10:59:43	1	9403
2785195	2013-01-05 11:06:51	1	9107
...	...	...	...
12330035	2013-01-08 12:06:00	2	9300
12445211	2013-01-08 21:39:00	1	10320
12541148	2013-01-09 14:03:00	1	10620
12612492	2013-01-09 18:29:00	1	10680
12626775	2013-01-09 15:23:00	1	9540
12664669	2013-01-09 19:53:00	1	10380
13063516	2013-01-11 15:59:00	5	10140
13160480	2013-01-11 22:46:00	5	10380
13184498	2013-01-02 18:59:47	2	9128
13237102	2013-01-04 18:16:00	1	9600
13367219	2013-01-12 18:34:00	1	9780
13540502	2013-01-10 12:44:53	1	9086
13544928	2013-01-02 04:39:37	1	9653
13546371	2013-01-02 02:53:36	1	9864
13653505	2013-01-08 22:15:33	1	9176
13655713	2013-01-11 19:22:40	1	9013
13684403	2013-01-24 12:55:35	1	9369
14151908	2013-01-27 14:41:23	3	9137
14171902	2013-01-27 18:58:10	2	10265
14216991	2013-01-23 10:23:55	1	9970
14241138	2013-01-21 19:56:53	4	9172
14351954	2013-01-21 19:36:58	1	9587
14373172	2013-01-22 10:11:26	1	9370
14409370	2013-01-23 20:18:26	1	9276
14549307	2013-01-26 12:29:43	1	9763
14669733	2013-01-06 18:43:29	1	9636
14669916	2013-01-06 23:46:39	1	9378
14695019	2013-01-07 11:28:10	1	9662
14739586	2013-01-01 16:14:38	1	9712
14751945	2013-01-08 15:10:20	1	9742

	trip_distance	pickup_longitude	pickup_latitude	dropoff_longitude	\
29526	0.00	-73.978409	40.750923	-73.978363	
113545	30.76	-73.984917	40.760132	-73.782494	
157147	0.06	0.000000	0.000000	0.000000	
256114	38.91	-73.783791	40.648682	-73.782234	
486795	37.88	-73.974350	40.764843	-73.965103	
900367	0.04	-73.794266	40.656326	-73.794266	
933085	18.61	-73.996048	40.724380	-73.998268	
958662	34.46	-73.950569	40.779404	-73.980820	
958809	30.38	-73.978355	40.756683	-73.987038	
1196135	36.30	-73.776970	40.646255	-73.782036	
1211812	0.75	-73.983330	40.777264	-73.981964	
1310253	0.00	-73.997971	40.755756	-73.997963	
1570703	18.21	0.000000	0.000000	0.000000	

1720244	61.40	-73.780708	40.646400	-74.687805
1844957	51.70	-73.711945	40.384571	-73.789993
1879615	37.04	-73.985077	40.746101	-73.789795
1894145	31.16	-73.991875	40.749706	-73.790199
1912151	84.22	-73.781654	40.644852	-73.785782
2149604	38.84	-73.794716	40.656471	-73.986053
2481328	11.20	-73.954407	40.781284	-73.951660
2514809	20.60	-74.013206	40.710148	-73.786308
2523595	10.40	-73.975586	40.758331	-73.985619
2541881	2.40	-73.989311	40.732903	-73.957008
2573058	22.10	-73.782036	40.644596	-73.976288
2574328	18.70	-73.983971	40.759174	-73.982414
2592899	4.60	-73.977364	40.755661	-73.981339
2608234	2.80	-73.963417	40.772125	-74.002029
2653124	0.60	-74.002190	40.734669	-73.977234
2785138	74.80	-73.810341	40.691788	-73.966118
2785195	52.50	-73.780769	40.645142	-73.969398
...	...	...	...	...
12330035	0.66	-73.993881	40.738701	-73.988579
12445211	9.13	0.000000	0.000000	0.000000
12541148	0.00	0.000000	0.000000	0.000000
12612492	5.43	-73.923073	40.743893	-73.931831
12626775	7.44	-73.794334	40.657047	-73.740540
12664669	4.79	-73.789345	40.645210	-73.739288
13063516	10.16	-73.869965	40.733578	-73.938148
13160480	54.39	-74.127823	41.316471	-73.989288
13184498	19.80	-73.978500	40.756199	-73.955475
13237102	48.51	-73.786026	40.642048	-73.789833
13367219	60.02	-73.786011	40.641579	-73.944458
13540502	18.70	-73.782013	40.644684	-73.987839
13544928	73.90	-73.991295	40.750282	-73.929619
13546371	76.60	-73.989502	40.757202	-73.834541
13653505	2.60	-73.784920	40.765060	-73.784630
13655713	19.50	-73.786026	40.642086	-73.987785
13684403	10.80	-74.009361	40.715294	-73.870720
14151908	38.30	-73.786896	40.641644	-73.788910
14171902	40.50	-73.984901	40.760071	-73.985092
14216991	1.70	-73.997025	40.696365	-74.010002
14241138	55.70	-73.789864	40.655106	-74.454689
14351954	18.10	-73.789780	40.643726	-73.977318
14373172	2.40	-73.996933	40.747307	-73.978844
14409370	0.60	-73.965424	40.763363	-73.936668
14549307	9.60	-74.004036	40.742004	-73.966164
14669733	20.10	-73.873055	40.773914	-73.862984
14669916	5.50	-74.007202	40.717548	-73.960052
14695019	19.30	-73.985008	40.768497	-73.985023
14739586	11.80	-73.987633	40.760143	-73.751259
14751945	1.50	-73.987526	40.775787	-73.982376

	dropoff_latitude
29526	40.750961
113545	40.644135
157147	0.000000
256114	40.644333
486795	40.754967
900367	40.656326
933085	40.729496
958662	40.774658
958809	40.729141
1196135	40.644459
1211812	40.772846
1310253	40.755760
1570703	0.000000
1720244	40.601070
1844957	40.646767
1879615	40.643063
1894145	40.643837
1912151	40.651398
2149604	40.729328
2481328	40.781292
2514809	40.644272
2523595	40.759552
2541881	40.770420
2573058	40.753960
2574328	40.766602
2592899	40.786732
2608234	40.741413
2653124	40.749500
2785138	40.804138
2785195	40.760311
...	...
12330035	40.737381
12445211	0.000000
12541148	0.000000
12612492	40.744858
12626775	40.636597
12664669	40.633133
13063516	40.768436
13160480	40.757950
13184498	40.776093
13237102	40.643162
13367219	40.775414
13540502	40.759895
13544928	40.812065
13546371	40.766041
13653505	40.764587

```
13655713         40.767170
13684403         40.774055
14151908         40.641468
14171902         40.760303
14216991         40.720261
14241138         40.519939
14351954         40.758293
14373172         40.756138
14409370         40.758183
14549307         40.770756
14669733         40.769032
14669916         40.782143
14695019         40.768528
14739586         40.739983
14751945         40.773479

[169 rows x 14 columns]
```

```
In [20]: df=df[(~df["medallion"].isnull()) & (~df["passenger_count"].isnull()) & (df["passenge
```

```
In [21]: df.shape
```

```
Out[21]: (12812181, 14)
```

4. Identificar si existen renglones con errores en los datos, por ejemplo si hay columnas de mas (o de menos), si hay campos vacios, etc. Si se detectan renglones con errores:

- indicar claramente cuantos son los renglones con errores y a que archivos corresponden 1964434
- crear un nuevo conjunto de datos con los errores eliminados

```
In [22]: %time df['trip_distance'].mean()
```

```
CPU times: user 24 ms, sys: 4 ms, total: 28 ms
Wall time: 44.6 ms
```

```
Out[22]: 2.7635248565415234
```

5. Crear un DataFrame usando la libreria pandas y responder lo siguiente:

- Indicar el tiempo en segundos, que tarda la libreria pandas en leer un archivo a un DataFrame Wall time: 1min 4s
- Cual es la huella de memoria del proceso usado para generar el objeto DataFrame Huella de memoria: 3225.76171875 MB
- Cuanta memoria del sistema se usa para crear el objeto anterior Memoria del sistema: 1578.312759399414 MB
- Indicar el tiempo que tarda pandas en obtener el promedio de la distancia de viaje (trip_distance) Wall time: 44.6 ms