Schools_scores

August 30, 2019

1 Schools

[1]:
```
import pandas as pd
import numpy as np
df = pd.read_csv("SchoolTable.csv")
```

[2]:
```
#df
```

[3]:
```
df['Overall_Rating1'] = df['Overall_Rating1'].replace("Level 1+",
                                                      5).replace("Level 1",
                                                      4).replace("Level 2+",
                                                      3).replace("Level 2",
                                                      2).replace("Level 3",1)
df['Overall_Rating1'] = df['Overall_Rating1'].astype('category').cat.codes
```

[4]:
```
df['Overall_Rating1'].dtype
```

[4]:
```
dtype('int8')
```

[5]:
```
type(df['Overall_Rating1'].iloc[[0,1,2,3,4]])
```

[5]:
```
pandas.core.series.Series
```

[9]:
```
len(df['Zip'].unique())
```

[9]:
```
51
```

[10]:
```
df['Zip'].dtype
```

[10]:
```
dtype('int64')
```

[11]:
```
df.groupby('Zip')['Overall_Rating1'].count()
```

[11]:
```
Zip
60602     2
60605     1
60607     6
60608    27
60609    27
60610     8
60612    25
60613     7
60614     7
```

```
60615     9
60616    16
60617    27
60618    19
60619    19
60620    22
60621    16
60622    16
60623    35
60624    19
60625    15
60626     7
60628    25
60629    21
60630     4
60631     7
60632    29
60633     2
60634    13
60636    14
60637    16
60638     9
60639    16
60640     6
60641    12
60642     7
60643    21
60644    15
60645     6
60646     4
60647    14
60649    10
60651    15
60652    11
60653    13
60655     5
60656     4
60657     6
60659     7
60660     5
60707     2
60827     5
Name: Overall_Rating1, dtype: int64
```

[12]:
```
df.groupby('Zip')['Overall_Rating1'].sum()
```

[12]:
```
Zip
60602     6
```

60605	4
60607	20
60608	74
60609	58
60610	25
60612	51
60613	25
60614	26
60615	21
60616	42
60617	70
60618	56
60619	48
60620	57
60621	30
60622	40
60623	85
60624	48
60625	53
60626	20
60628	50
60629	59
60630	15
60631	19
60632	89
60633	4
60634	41
60636	19
60637	39
60638	23
60639	41
60640	16
60641	32
60642	22
60643	46
60644	34
60645	22
60646	16
60647	37
60649	24
60651	42
60652	29
60653	28
60655	16
60656	11
60657	21
60659	23

```
60660    16
60707     5
60827     9
Name: Overall_Rating1, dtype: int8
```

2 Test

[13]:
```
df.loc[df['Zip'] == 60607]
```

[13]:
```
     IdInt                                        Long_Name Primary_Category  \
345    346                Mark Skinner Elementary School                ES
381    382                            STEM Magnet Academy               ES
502    503    Andrew Jackson Elementary Language Academy                ES
553    554                Chicago Virtual Charter School                HS
569    570  Galileo Math & Science Scholastic Academy ES                ES
599    600             Whitney M Young Magnet High School               HS

     School_Latitude  School_Longitude    Zip  Overall_Rating1
345        41.879103        -87.659419  60607                4
381        41.868576        -87.664969  60607                4
502        41.874300        -87.660985  60607                4
553        41.880550        -87.649772  60607                1
569        41.871255        -87.653366  60607                3
599        41.878603        -87.664233  60607                4
```

[ ]:
```

```

3 Per Capita

[14]:
```
scores = pd.DataFrame(df.groupby('Zip')['Overall_Rating1'].sum())
numbers = pd.DataFrame(df.groupby('Zip')['Overall_Rating1'].count())
```

[15]:
```
scores['zipcode'] = scores.index
numbers['zipcode'] = numbers.index
```

[16]:
```
scores.reset_index(drop = True, inplace=True)
numbers.reset_index(drop = True, inplace=True)
```

[17]:
```
df2 = pd.merge(scores, numbers, on='zipcode', how='outer')
df2 = df2[['zipcode', 'Overall_Rating1_x','Overall_Rating1_y']]
```

[18]:
```
df2.rename(columns={'zipcode':'Zip_code'}, inplace=True)
df2.rename(columns={'Overall_Rating1_x':'Total scores'}, inplace=True)
df2.rename(columns={'Overall_Rating1_y':'# of schools'}, inplace=True)
```

[19]:
```
pop = pd.read_csv('PopTable.csv')
```

[20]:
```
#pop
```

[21]:
```
ziplist = list(df['Zip'].unique())
```

[22]:
```
filterzip = pop.loc[pop['zip_code'].isin(ziplist)]
type(filterzip)
```

[22]:
```
pandas.core.frame.DataFrame
```

[50]:
```
#filterzip
```

[53]:
```
pop = pop[['zip_code','population']]
```

[66]:
```
#pop
```

[23]:
```
newfilter = filterzip.ix[:, 1:3]
newfilter.rename(columns={'zip_code':'Zip_code'}, inplace=True)
#newfilter
```

```
/Users/zhongyizhang/env/lib/python3.7/site-packages/ipykernel_launcher.py:1:
DeprecationWarning:
.ix is deprecated. Please use
.loc for label based indexing or
.iloc for positional indexing

See the documentation here:
http://pandas.pydata.org/pandas-docs/stable/indexing.html#ix-indexer-is-
deprecated
  """Entry point for launching an IPython kernel.
/Users/zhongyizhang/env/lib/python3.7/site-packages/pandas/core/frame.py:4025:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  return super(DataFrame, self).rename(**kwargs)
```

[24]:
```
df3 = pd.merge(df2, newfilter, on='Zip_code', how='outer')
df3['Per school'] = df3['Total scores']/df3['# of schools']
df3['Per capita'] = df3['Total scores']/df3['population']
```

[69]:
```
df3
```

[69]:

	Zip_code	Total scores	# of schools	population	Per school	Per capita
0	60602	6	2	1252	3.000000	0.004792
1	60605	4	1	26623	4.000000	0.000150
2	60607	20	6	28377	3.333333	0.000705
3	60608	74	27	78072	2.740741	0.000948
4	60609	58	27	62250	2.148148	0.000932
5	60610	25	8	38438	3.125000	0.000650
6	60612	51	25	35559	2.040000	0.001434
7	60613	25	7	49519	3.571429	0.000505
8	60614	26	7	69817	3.714286	0.000372

9	60615	21	9	40257	2.333333	0.000522
10	60616	42	16	52580	2.625000	0.000799
11	60617	70	27	80002	2.592593	0.000875
12	60618	56	19	95632	2.947368	0.000586
13	60619	48	19	62822	2.526316	0.000764
14	60620	57	22	69299	2.590909	0.000823
15	60621	30	16	31383	1.875000	0.000956
16	60622	40	16	54467	2.500000	0.000734
17	60623	85	35	88137	2.428571	0.000964
18	60624	48	19	38134	2.526316	0.001259
19	60625	53	15	79157	3.533333	0.000670
20	60626	20	7	50090	2.857143	0.000399
21	60628	50	25	68077	2.000000	0.000734
22	60629	59	21	115104	2.809524	0.000513
23	60630	15	4	57627	3.750000	0.000260
24	60631	19	7	28238	2.714286	0.000673
25	60632	89	29	91668	3.068966	0.000971
26	60633	4	2	12817	2.000000	0.000312
27	60634	41	13	73382	3.153846	0.000559
28	60636	19	14	35779	1.357143	0.000531
29	60637	39	16	49158	2.437500	0.000793
30	60638	23	9	57746	2.555556	0.000398
31	60639	41	16	90211	2.562500	0.000454
32	60640	16	6	67088	2.666667	0.000238
33	60641	32	12	70642	2.666667	0.000453
34	60642	22	7	19508	3.142857	0.001128
35	60643	46	21	50507	2.190476	0.000911
36	60644	34	15	49645	2.266667	0.000685
37	60645	22	6	47131	3.666667	0.000467
38	60646	16	4	27454	4.000000	0.000583
39	60647	37	14	88866	2.642857	0.000416
40	60649	24	10	45218	2.400000	0.000531
41	60651	42	15	61759	2.800000	0.000680
42	60652	29	11	43228	2.636364	0.000671
43	60653	28	13	31045	2.153846	0.000902
44	60655	16	5	28741	3.200000	0.000557
45	60656	11	4	27926	2.750000	0.000394
46	60657	21	6	70105	3.500000	0.000300
47	60659	23	7	38995	3.285714	0.000590
48	60660	16	5	41490	3.200000	0.000386
49	60707	5	2	43451	2.500000	0.000115
50	60827	9	5	28864	1.800000	0.000312

[26]:
```
dfzip = pd.read_csv("Zip Property Values.csv")
```

[47]:
```
df4 = dfzip['zip_code']
df4 = pd.DataFrame(df4)
df4.rename(columns={'zip_code':'Zip_code'}, inplace=True)
```

```
df5 = pd.merge(df4[['Zip_code']],df3[['Zip_code','Total scores', '# of schools',
                                      'population','Per school','Per capita']],
               on='Zip_code',how='left')
```

[57]:
```
df5= df5.drop(columns=['population'])
```

[64]:
```
pop.rename(columns={'zip_code':'Zip_code'}, inplace=True)
df6 = pd.merge(df5[['Zip_code','Total scores', '# of schools','Per school',
                    'Per capita']],pop[['Zip_code','population']],
               on='Zip_code',how='left')
```

[70]:
```
df6 = df6.reindex(columns=['Zip_code','Total scores','# of␣
↪schools','population',
                           'Per school','Per capita'])
```

[72]:
```
df6=df6.fillna(0)
```

[73]:
```
df6
```

[73]:

	Zip_code	Total scores	# of schools	population	Per school	Per capita
0	60601	0.0	0.0	13695	0.000000	0.000000
1	60602	6.0	2.0	1252	3.000000	0.004792
2	60603	0.0	0.0	1029	0.000000	0.000000
3	60604	0.0	0.0	619	0.000000	0.000000
4	60605	4.0	1.0	26623	4.000000	0.000150
5	60606	0.0	0.0	3011	0.000000	0.000000
6	60607	20.0	6.0	28377	3.333333	0.000705
7	60608	74.0	27.0	78072	2.740741	0.000948
8	60609	58.0	27.0	62250	2.148148	0.000932
9	60610	25.0	8.0	38438	3.125000	0.000650
10	60611	0.0	0.0	31563	0.000000	0.000000
11	60612	51.0	25.0	35559	2.040000	0.001434
12	60613	25.0	7.0	49519	3.571429	0.000505
13	60614	26.0	7.0	69817	3.714286	0.000372
14	60615	21.0	9.0	40257	2.333333	0.000522
15	60616	42.0	16.0	52580	2.625000	0.000799
16	60617	70.0	27.0	80002	2.592593	0.000875
17	60618	56.0	19.0	95632	2.947368	0.000586
18	60619	48.0	19.0	62822	2.526316	0.000764
19	60620	57.0	22.0	69299	2.590909	0.000823
20	60621	30.0	16.0	31383	1.875000	0.000956
21	60622	40.0	16.0	54467	2.500000	0.000734
22	60623	85.0	35.0	88137	2.428571	0.000964
23	60624	48.0	19.0	38134	2.526316	0.001259
24	60625	53.0	15.0	79157	3.533333	0.000670
25	60626	20.0	7.0	50090	2.857143	0.000399
26	60628	50.0	25.0	68077	2.000000	0.000734
27	60629	59.0	21.0	115104	2.809524	0.000513
28	60630	15.0	4.0	57627	3.750000	0.000260
29	60631	19.0	7.0	28238	2.714286	0.000673

..	...	...	...	...	...	...
34	60637	39.0	16.0	49158	2.437500	0.000793
35	60638	23.0	9.0	57746	2.555556	0.000398
36	60639	41.0	16.0	90211	2.562500	0.000454
37	60640	16.0	6.0	67088	2.666667	0.000238
38	60641	32.0	12.0	70642	2.666667	0.000453
39	60642	22.0	7.0	19508	3.142857	0.001128
40	60643	46.0	21.0	50507	2.190476	0.000911
41	60644	34.0	15.0	49645	2.266667	0.000685
42	60645	22.0	6.0	47131	3.666667	0.000467
43	60646	16.0	4.0	27454	4.000000	0.000583
44	60647	37.0	14.0	88866	2.642857	0.000416
45	60649	24.0	10.0	45218	2.400000	0.000531
46	60651	42.0	15.0	61759	2.800000	0.000680
47	60652	29.0	11.0	43228	2.636364	0.000671
48	60653	28.0	13.0	31045	2.153846	0.000902
49	60654	0.0	0.0	17328	0.000000	0.000000
50	60655	16.0	5.0	28741	3.200000	0.000557
51	60656	11.0	4.0	27926	2.750000	0.000394
52	60657	21.0	6.0	70105	3.500000	0.000300
53	60659	23.0	7.0	38995	3.285714	0.000590
54	60660	16.0	5.0	41490	3.200000	0.000386
55	60661	0.0	0.0	9343	0.000000	0.000000
56	60706	0.0	0.0	23604	0.000000	0.000000
57	60707	5.0	2.0	43451	2.500000	0.000115
58	60712	0.0	0.0	12637	0.000000	0.000000
59	60714	0.0	0.0	29730	0.000000	0.000000
60	60803	0.0	0.0	22762	0.000000	0.000000
61	60804	0.0	0.0	83972	0.000000	0.000000
62	60805	0.0	0.0	19849	0.000000	0.000000
63	60827	9.0	5.0	28864	1.800000	0.000312

[64 rows x 6 columns]

[74]:
```
df6.to_csv('Schools_scores_2.csv')
```

[ ]:

[ ]:

[ ]: