revise2

March 12, 2024

[ ]:
```
import pandas as pd
```

[ ]:
```
df = pd.read_csv('melb_data.csv')
```

[ ]:
```
print(df.size)
print(df.shape)
print(df.ndim)
print(df.info())
```

```
285180
(13580, 21)
2
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13580 entries, 0 to 13579
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Suburb         13580 non-null  object
 1   Address        13580 non-null  object
 2   Rooms          13580 non-null  int64
 3   Type           13580 non-null  object
 4   Price          13580 non-null  float64
 5   Method         13580 non-null  object
 6   SellerG        13580 non-null  object
 7   Date           13580 non-null  object
 8   Distance       13580 non-null  float64
 9   Postcode       13580 non-null  float64
 10  Bedroom2       13580 non-null  float64
 11  Bathroom       13580 non-null  float64
 12  Car            13518 non-null  float64
 13  Landsize       13580 non-null  float64
 14  BuildingArea   7130 non-null   float64
 15  YearBuilt      8205 non-null   float64
 16  CouncilArea    12211 non-null  object
 17  Lattitude      13580 non-null  float64
 18  Longtitude     13580 non-null  float64
 19  Regionname     13580 non-null  object
 20  Propertycount  13580 non-null  float64
```

```
dtypes: float64(12), int64(1), object(8)
memory usage: 2.2+ MB
None
```

[ ]:
```
df.describe().loc[['min','max','mean','std']]
```

[ ]:
```
          Rooms         Price   Distance     Postcode   Bedroom2  Bathroom  \
min    1.000000  8.500000e+04   0.000000  3000.000000   0.000000  0.000000
max   10.000000  9.000000e+06  48.100000  3977.000000  20.000000  8.000000
mean   2.937997  1.075684e+06  10.137776  3105.301915   2.914728  1.534242
std    0.955748  6.393107e+05   5.868725    90.676964   0.965921  0.691712

             Car       Landsize  BuildingArea    YearBuilt  Lattitude  \
min    0.000000       0.000000      0.000000  1196.000000 -38.182550
max   10.000000  433014.000000  44515.000000  2018.000000 -37.408530
mean   1.610075     558.416127    151.967650  1964.684217 -37.809203
std    0.962634    3990.669241    541.014538    37.273762   0.079260

      Longtitude  Propertycount
min   144.431810     249.000000
max   145.526350   21650.000000
mean  144.995216    7454.417378
std     0.103916    4378.581772
```

[ ]:
```
df.describe().
  ↪loc[['min','max','mean','std'],['Price','Landsize','Propertycount']]
```

[ ]:
```
             Price       Landsize  Propertycount
min   8.500000e+04       0.000000     249.000000
max   9.000000e+06  433014.000000   21650.000000
mean  1.075684e+06     558.416127    7454.417378
std   6.393107e+05    3990.669241    4378.581772
```

[ ]:
```
df.loc[df['Landsize']<500].describe().loc[['min','max','mean','std']]
```

[ ]:
```
          Rooms         Price   Distance     Postcode   Bedroom2  Bathroom  \
min    1.000000  8.500000e+04   0.000000  3000.000000   0.000000  0.000000
max   10.000000  5.700000e+06  47.300000  3977.000000  10.000000  7.000000
mean   2.587798  9.484325e+05   8.131710  3098.894210   2.566558  1.429654
std    0.822415  5.065054e+05   4.713157    79.210585   0.817484  0.597363

             Car    Landsize  BuildingArea    YearBuilt  Lattitude  Longtitude  \
min    0.000000    0.000000      0.000000  1850.000000 -38.164390  144.568870
max    7.000000  499.000000   1561.000000  2018.000000 -37.491750  145.453760
mean   1.288069  197.216585    119.261818  1965.291500 -37.810351  144.982682
std    0.748220  155.069985     66.255755    41.101236   0.065692    0.079523

      Propertycount
```

```
min      394.000000
max    21650.000000
mean    7712.066694
std     4328.118170
```

[ ]:
```
df.loc[(df['Bedroom2']==2) & (df['Bathroom'] == 1) & (df['Car'] == 1)].
↪describe().loc[['min','max','mean','std']]
```

[ ]:
```
          Rooms         Price   Distance     Postcode  Bedroom2  Bathroom  Car  \
min    1.000000  1.450000e+05   0.000000  3000.000000       2.0       1.0  1.0
max    4.000000  2.905000e+06  41.000000  3910.000000       2.0       1.0  1.0
mean   2.026205  6.794727e+05   8.056575  3100.423023       2.0       1.0  1.0
std    0.194171  2.908005e+05   4.304758    67.580018       0.0       0.0  0.0

           Landsize  BuildingArea    YearBuilt  Lattitude  Longtitude  \
min        0.000000      0.000000  1830.000000 -38.164390  144.571590
max    17200.000000   1143.000000  2016.000000 -37.570630  145.335010
mean     348.408985     84.769562  1969.518362 -37.811354  144.991409
std      963.376076     44.334332    31.775777   0.063619    0.072949

      Propertycount
min      438.000000
max    21650.000000
mean    7854.260178
std     4576.352268
```

[ ]:
```
remove_row= df.dropna(axis = 0)
print(remove_row.shape)
print(remove_row.info())
```

```
(6196, 21)
<class 'pandas.core.frame.DataFrame'>
Index: 6196 entries, 1 to 12212
Data columns (total 21 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Suburb         6196 non-null   object
 1   Address        6196 non-null   object
 2   Rooms          6196 non-null   int64
 3   Type           6196 non-null   object
 4   Price          6196 non-null   float64
 5   Method         6196 non-null   object
 6   SellerG        6196 non-null   object
 7   Date           6196 non-null   object
 8   Distance       6196 non-null   float64
 9   Postcode       6196 non-null   float64
 10  Bedroom2       6196 non-null   float64
 11  Bathroom       6196 non-null   float64
```

```
 12  Car            6196 non-null   float64
 13  Landsize       6196 non-null   float64
 14  BuildingArea   6196 non-null   float64
 15  YearBuilt      6196 non-null   float64
 16  CouncilArea    6196 non-null   object
 17  Lattitude      6196 non-null   float64
 18  Longtitude     6196 non-null   float64
 19  Regionname     6196 non-null   object
 20  Propertycount  6196 non-null   float64
dtypes: float64(12), int64(1), object(8)
memory usage: 1.0+ MB
None
```

[ ]:
```
replace_zero = df.fillna(0)
print(df.describe().loc[['mean']])
print(replace_zero.describe().loc[['mean']])
# you should not use this approach because it will make the data deviated from␣
 ↪the original
```

```
          Rooms         Price   Distance     Postcode  Bedroom2  Bathroom  \
mean  2.937997  1.075684e+06  10.137776  3105.301915  2.914728  1.534242

           Car    Landsize  BuildingArea    YearBuilt  Lattitude  Longtitude  \
mean  1.610075  558.416127     151.96765  1964.684217 -37.809203  144.995216

      Propertycount
mean    7454.417378
          Rooms         Price   Distance     Postcode  Bedroom2  Bathroom  \
mean  2.937997  1.075684e+06  10.137776  3105.301915  2.914728  1.534242

           Car    Landsize  BuildingArea    YearBuilt  Lattitude  Longtitude  \
mean  1.602725  558.416127     79.788611  1187.056996 -37.809203  144.995216

      Propertycount
mean    7454.417378
```

[ ]:
```
print(df.describe().loc['mean'])
imputation = df.select_dtypes(include = 'number').columns
df[imputation] = df[imputation].fillna(df[imputation].median())
df.dropna(axis=0)
print(df.describe().loc['mean'])
# this should be used because the data mean still be the same
```

```
Rooms           2.937997e+00
Price           1.075684e+06
Distance        1.013778e+01
Postcode        3.105302e+03
Bedroom2        2.914728e+00
Bathroom        1.534242e+00
```

```
Car              1.611856e+00
Landsize         5.584161e+02
BuildingArea     1.396340e+02
YearBuilt        1.966788e+03
Lattitude       -3.780920e+01
Longtitude       1.449952e+02
Propertycount    7.454417e+03
Name: mean, dtype: float64
Rooms            2.937997e+00
Price            1.075684e+06
Distance         1.013778e+01
Postcode         3.105302e+03
Bedroom2         2.914728e+00
Bathroom         1.534242e+00
Car              1.611856e+00
Landsize         5.584161e+02
BuildingArea     1.396340e+02
YearBuilt        1.966788e+03
Lattitude       -3.780920e+01
Longtitude       1.449952e+02
Propertycount    7.454417e+03
Name: mean, dtype: float64
```

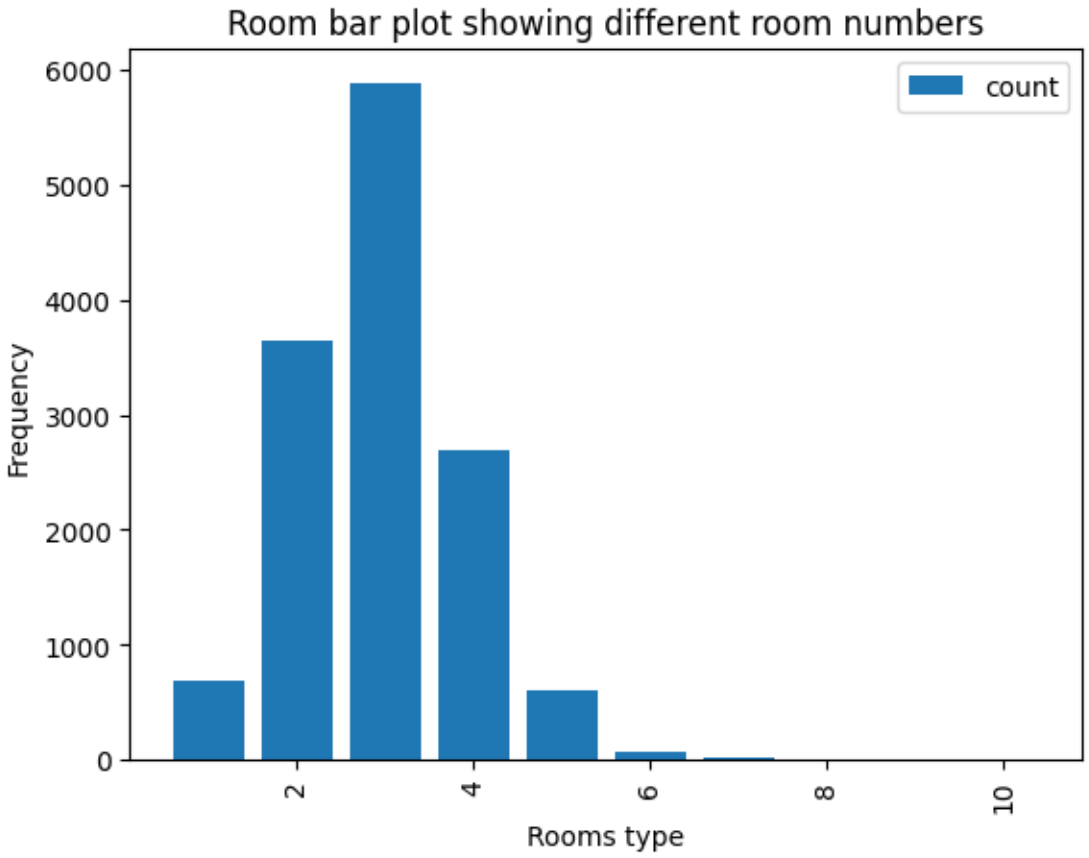
[ ]:
```
df['Date']=pd.to_datetime(df['Date'],format = 'mixed')
df['Date'] =df['Date'].dt.strftime(date_format='%Y/%m/%d')
df['Date']
```

[ ]:
```
0        2016/03/12
1        2016/04/02
2        2017/04/03
3        2017/04/03
4        2016/04/06
            …
13575    2017/08/26
13576    2017/08/26
13577    2017/08/26
13578    2017/08/26
13579    2017/08/26
Name: Date, Length: 13580, dtype: object
```

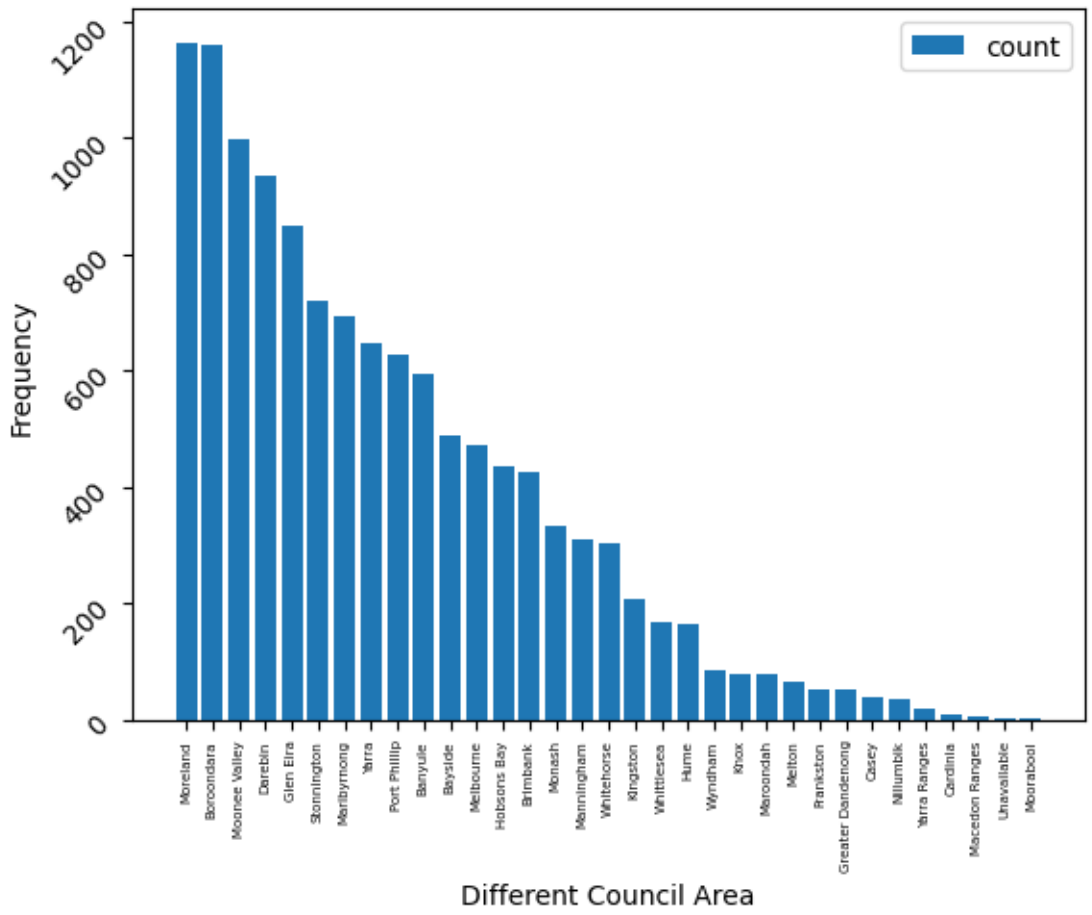
[ ]:
```
import matplotlib.pyplot as plt

plt.bar(height=df['Rooms'].value_counts(),x=df['Rooms'].value_counts().
  ↪index,label='count')
plt.xlabel('Rooms type')
plt.ylabel('Frequency')
plt.title('Room bar plot showing different room numbers')
```

```
plt.xticks(rotation='vertical')
plt.legend()
plt.show()
```



[ ]:
```
plt.bar(height=df['CouncilArea'].value_counts(),x=df['CouncilArea'].
  ↪value_counts().index,label='count')
plt.xlabel('Different Council Area')
plt.ylabel('Frequency')
plt.xticks(rotation=90,size=5)
plt.yticks(rotation=45)
plt.legend()
plt.show()
```



[ ]:
```
df.
↪groupby(['Regionname','Type'])[['Price','Bedroom2','Bathroom','Car','Landsize']].
↪sum()
```

[ ]:

Regionname	Type	Price	Bedroom2	Bathroom	Car
Eastern Metropolitan	h	1.404609e+09	4100.0	2062.0	2217.0
	t	1.026152e+08	353.0	214.0	193.0
	u	1.168766e+08	421.0	223.0	228.0
Eastern Victoria	h	3.571498e+07	172.0	93.0	105.0
	u	1.384000e+06	8.0	3.0	4.0
Northern Metropolitan	h	2.812445e+09	8310.0	3867.0	4371.0
	t	2.301298e+08	773.0	492.0	403.0
	u	4.513107e+08	1553.0	962.0	894.0
Northern Victoria	h	2.438800e+07	146.0	76.0	79.0
South-Eastern Metropolitan	h	3.709085e+08	1353.0	654.0	826.0
	t	2.283175e+07	71.0	46.0	42.0
	u	2.158450e+07	86.0	46.0	52.0

```
Southern Metropolitan      h     4.903898e+09    9065.0    4955.0  4931.0
                           t     5.122969e+08    1253.0     828.0   735.0
                           u     1.029868e+09    2993.0    1842.0  1746.0
Western Metropolitan       h     2.177255e+09    7259.0    3495.0  4162.0
                           t     1.723073e+08     685.0     436.0   360.0
                           u     2.046456e+08     872.0     494.0   482.0
Western Victoria           h     1.272075e+07     109.0      47.0    59.0

                                    Landsize
Regionname                 Type
Eastern Metropolitan       h      841537.0
                           t       31794.0
                           u       59480.0
Eastern Victoria           h      155448.0
                           u         886.0
Northern Metropolitan      h     1705412.0
                           t       97419.0
                           u      410377.0
Northern Victoria          h      137574.0
South-Eastern Metropolitan h      257751.0
                           t        5304.0
                           u       13241.0
Southern Metropolitan      h     1550001.0
                           t      118515.0
                           u      722423.0
Western Metropolitan       h     1163053.0
                           t       58450.0
                           u      233650.0
Western Victoria           h       20976.0
```

[ ]: