

Ben Deatsman

Classwork 3c

In [15]:

```
import pandas as pd
df = pd.read_csv('ames_raw.csv')
df
```

Out[15]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Lot Shape	Land Contour	...	Pool Area	Pool QC	Fence	Misc Feature	Misc Val	Mo Sold	Yr Sold	Sale Type	Sale Condition	SalePrice	
	0	1	526301100	20	RL	141.0	31770	Pave	NaN	IR1	Lvl	...	0	NaN	NaN	NaN	0	5	2010	WD	Normal	215000
	1	2	526350040	20	RH	80.0	11622	Pave	NaN	Reg	Lvl	...	0	NaN	MnPrv	NaN	0	6	2010	WD	Normal	105000
	2	3	526351010	20	RL	81.0	14267	Pave	NaN	IR1	Lvl	...	0	NaN	NaN	Gar2	12500	6	2010	WD	Normal	172000
	3	4	526353030	20	RL	93.0	11160	Pave	NaN	Reg	Lvl	...	0	NaN	NaN	NaN	0	4	2010	WD	Normal	244000
	4	5	527105010	60	RL	74.0	13830	Pave	NaN	IR1	Lvl	...	0	NaN	MnPrv	NaN	0	3	2010	WD	Normal	189900

	2925	2926	923275080	80	RL	37.0	7937	Pave	NaN	IR1	Lvl	...	0	NaN	GdPrv	NaN	0	3	2006	WD	Normal	142500
	2926	2927	923276100	20	RL	NaN	8885	Pave	NaN	IR1	Low	...	0	NaN	MnPrv	NaN	0	6	2006	WD	Normal	131000
	2927	2928	923400125	85	RL	62.0	10441	Pave	NaN	Reg	Lvl	...	0	NaN	MnPrv	Shed	700	7	2006	WD	Normal	132000
	2928	2929	924100070	20	RL	77.0	10010	Pave	NaN	Reg	Lvl	...	0	NaN	NaN	NaN	0	4	2006	WD	Normal	170000
	2929	2930	924151050	60	RL	74.0	9627	Pave	NaN	Reg	Lvl	...	0	NaN	NaN	NaN	0	11	2006	WD	Normal	188000

2930 rows × 82 columns

In [24]:

```
df.columns
```

Out[24]:

Index(['Order', 'PID', 'MS SubClass', 'MS Zoning', 'Lot Frontage', 'Lot Area', 'Street', 'Alley', 'Lot Shape', 'Land Contour', 'Utilities', 'Lot Config', 'Land Slope', 'Neighborhood', 'Condition 1', 'Condition 2', 'Bldg Type', 'House Style', 'Overall Qual', 'Overall Cond', 'Year Built', 'Year Remod/Add', 'Roof Style', 'Roof Matl', 'Exterior 1st', 'Exterior 2nd', 'Mas Vnr Type', 'Mas Vnr Area', 'Exter Qual', 'Exter Cond', 'Foundation', 'Bsmt Qual', 'Bsmt Cond', 'Bsmt Exposure', 'BsmtFin Type 1', 'BsmtFin SF 1', 'BsmtFin Type 2', 'BsmtFin SF 2', 'Bsmt Unf SF', 'Total Bsmt SF', 'Heating', 'Heating QC', 'Central Air', 'Electrical', '1st Flr SF', '2nd Flr SF', 'Low Qual Fin SF', 'Gr Liv Area', 'Bsmt Full Bath', 'Bsmt Half Bath', 'Full Bath', 'Half Bath', 'Bedroom AbvGr', 'Kitchen AbvGr', 'Kitchen Qual', 'TotRms AbvGrd', 'Functional', 'Fireplaces', 'Fireplace Qu', 'Garage Type', 'Garage Yr Blt', 'Garage Finish', 'Garage Cars', 'Garage Area', 'Garage Qual', 'Garage Cond', 'Paved Drive', 'Wood Deck SF', 'Open Porch SF', 'Enclosed Porch', '3Ssn Porch', 'Screen Porch', 'Pool Area', 'Pool QC', 'Fence', 'Misc Feature', 'Misc Val', 'Mo Sold', 'Yr Sold', 'Sale Type', 'Sale Condition', 'SalePrice'], dtype='object')

In [94]:

```
#Highest Neighborhood Median SalesPrice: Veenker
df.groupby('Neighborhood', as_index=False).agg({'SalePrice': 'median'}).max()
```

Out[94]:

Neighborhood Veenker
SalePrice 319000.0
dtype: object

In [104...]:

```
df.groupby('Bedroom AbvGr', as_index=False).agg({'SalePrice': ['median', 'mean']})
```

Out[104...]:

	Bedroom AbvGr	SalePrice	
		median	mean
0	0	202500.0	218494.875000
1	1	167750.0	183017.339286
2	2	140000.0	162167.678331
3	3	165000.0	179711.767063
4	4	189500.0	216357.045000
5	5	182000.0	206244.250000
6	6	145000.0	159701.714286
7	8	200000.0	200000.000000

In [72]:

```
#Veenker has the highest median sales price for 3 bedroom homes
numbed = df['Bedroom AbvGr'] == 3
df[numbed].groupby('Neighborhood', as_index=False).agg({'SalePrice': 'median'}).max()
```

Out[72]:

Neighborhood Veenker
SalePrice 377426.0
dtype: object

In [76]:

```
#Blueste has the lowest median sales price for 3 bedroom homes
numbed = df['Bedroom AbvGr'] == 3
df[numbed].groupby('Neighborhood', as_index=False).agg({'SalePrice': 'median'}).min()
```

Out[76]:

Neighborhood Blueste
SalePrice 87775.0
dtype: object

In [88]:

```
price_per_sqft = df['SalePrice'] / df['Gr Liv Area']
df['price_per_sqft'] = price_per_sqft
```

In [96]:

```
#This does not differ from solution 1. It tells that when the Sale Price is higher, the Price per SqFt is higher.
df.groupby('Neighborhood', as_index=False).agg({'price_per_sqft': 'median'}).max()
```

Out[96]:

Neighborhood Veenker
price_per_sqft 198.656617
dtype: object

In []: