Python Statistics

Loading Data

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib
import seaborn as sns
from sklearn import linear_model, model_selection, preprocessing
import xgboost as xgb
from scipy import stats
import matplotlib.pyplot as plt
```

In [2]:
```
df = pd.read_csv('housing_sale_data.csv', engine='pyarrow', dtype_backend='p
```

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

Out[3]: (2930, 82)

In [4]:
```
df.head()
```

Out[4]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Lo Shap
0	1	526301100	20	RL	141	31770	Pave	<NA>	IR
1	2	526350040	20	RH	80	11622	Pave	<NA>	Re
2	3	526351010	20	RL	81	14267	Pave	<NA>	IR
3	4	526353030	20	RL	93	11160	Pave	<NA>	Re
4	5	527105010	60	RL	74	13830	Pave	<NA>	IR

5 rows × 82 columns

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

Out[5]:

	Order	PID	MS SubClass	Lot Frontage	Lot Area	Ove Q
count	2930.0	2930.0	2930.0	2440.0	2930.0	293
mean	1465.5	714464496.988737	57.387372	69.22459	10147.921843	6.094
std	845.96247	188730844.64939	42.638025	23.365335	7880.017759	1.411
min	1.0	526301100.0	20.0	21.0	1300.0	
25%	733.25	528477022.5	20.0	58.0	7440.25	
50%	1465.5	535453620.0	50.0	68.0	9436.5	
75%	2197.75	907181097.5	70.0	80.0	11555.25	
max	2930.0	1007100110.0	190.0	313.0	215245.0	1

8 rows × 39 columns

In [6]:

```
df.dtypes
```

Out[6]:

```
Order                int64[pyarrow]
PID                  int64[pyarrow]
MS SubClass          int64[pyarrow]
MS Zoning           string[pyarrow]
Lot Frontage         int64[pyarrow]
                          ...
Mo Sold              int64[pyarrow]
Yr Sold              int64[pyarrow]
Sale Type           string[pyarrow]
Sale Condition      string[pyarrow]
SalePrice            int64[pyarrow]
Length: 82, dtype: object
```

- Strings and Categories

In [7]:

```
# Compute and interpret summary statistics for categorical columns using the
df.select_dtypes('string').describe().T
```

Out[7]:

	count	unique	top	freq
MS Zoning	2930	7	RL	2273
Street	2930	2	Pave	2918
Alley	198	2	Grvl	120
Lot Shape	2930	4	Reg	1859
Land Contour	2930	4	Lvl	2633
Utilities	2930	3	AllPub	2927
Lot Config	2930	5	Inside	2140
Land Slope	2930	3	Gtl	2789
Neighborhood	2930	28	NAmes	443
Condition 1	2930	9	Norm	2522
Condition 2	2930	8	Norm	2900
Bldg Type	2930	5	1Fam	2425
House Style	2930	8	1Story	1481
Roof Style	2930	6	Gable	2321
Roof Matl	2930	8	CompShg	2887
Exterior 1st	2930	16	VinylSd	1026
Exterior 2nd	2930	17	VinylSd	1015
Mas Vnr Type	1155	4	BrkFace	880
Exter Qual	2930	4	TA	1799
Exter Cond	2930	5	TA	2549
Foundation	2930	6	PConc	1310
Bsmt Qual	2850	5	TA	1283
Bsmt Cond	2850	5	TA	2616
Bsmt Exposure	2847	4	No	1906
BsmtFin Type 1	2850	6	GLQ	859
BsmtFin Type 2	2849	6	Unf	2499
Heating	2930	6	GasA	2885
Heating QC	2930	5	Ex	1495
Central Air	2930	2	Y	2734
Electrical	2929	5	SBrkr	2682
Kitchen Qual	2930	5	TA	1494
Functional	2930	8	Typ	2728
Fireplace Qu	1508	5	Gd	744

	count	unique	top	freq
Garage Type	2773	6	Attchd	1731
Garage Finish	2771	3	Unf	1231
Garage Qual	2771	5	TA	2615
Garage Cond	2771	5	TA	2665
Paved Drive	2930	3	Y	2652
Pool QC	13	4	Ex	4
Fence	572	4	MnPrv	330
Misc Feature	106	5	Shed	95
Sale Type	2930	10	WD	2536
Sale Condition	2930	6	Normal	2413

In [ ]:

```
```

In [8]:

```
# Convert string columns to the ``'category'`` data type to save memory.
(df
 .select_dtypes('string')
 .memory_usage(deep=True)
 .sum()
)
```

Out[8]: np.int64(929599)

In [9]:

```
(df
 .select_dtypes('string')
 .astype('category')
 .memory_usage(deep=True)
 .sum()
)
```

Out[9]: np.int64(137945)

In [10]:

```
# Missing numeric columns
(df
 .isna()
 .mean()
 .mul(100)
 .pipe(lambda ser: ser[ser > 0])
)
```

Out[10]:
```
Lot Frontage      16.723549
Alley             93.242321
Mas Vnr Type      60.580205
Mas Vnr Area       0.784983
Bsmt Qual          2.730375
Bsmt Cond          2.730375
Bsmt Exposure      2.832765
BsmtFin Type 1     2.730375
BsmtFin SF 1       0.034130
BsmtFin Type 2     2.764505
BsmtFin SF 2       0.034130
Bsmt Unf SF        0.034130
Total Bsmt SF      0.034130
Electrical         0.034130
Bsmt Full Bath     0.068259
Bsmt Half Bath     0.068259
Fireplace Qu      48.532423
Garage Type        5.358362
Garage Yr Blt      5.426621
Garage Finish      5.426621
Garage Cars        0.034130
Garage Area        0.034130
Garage Qual        5.426621
Garage Cond        5.426621
Pool QC           99.556314
Fence             80.477816
Misc Feature      96.382253
dtype: float64
```

In [11]:
```
# Missing string values
(df
 .query('`Pool QC`.isna()')
)
```

Out[11]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
0	1	526301100	20	RL	141	31770	Pave	<NA>	
1	2	526350040	20	RH	80	11622	Pave	<NA>	
2	3	526351010	20	RL	81	14267	Pave	<NA>	
3	4	526353030	20	RL	93	11160	Pave	<NA>	
4	5	527105010	60	RL	74	13830	Pave	<NA>	
...	...	...	...	...	...	...	...	...	
2925	2926	923275080	80	RL	37	7937	Pave	<NA>	
2926	2927	923276100	20	RL	<NA>	8885	Pave	<NA>	
2927	2928	923400125	85	RL	62	10441	Pave	<NA>	
2928	2929	924100070	20	RL	77	10010	Pave	<NA>	
2929	2930	924151050	60	RL	74	9627	Pave	<NA>	

2917 rows × 82 columns

In [12]:

```
(df
 .query('`Pool QC` == "NA"')
)
```

Out[12]:

Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Lot Shape	Lan Contou

0 rows × 82 columns

In [13]:

```
# Fill in empty string with 'Not Applicable'
(df
 .assign(
     **df.select_dtypes('string').replace('', 'Not Applicable'))
)
```

Out[13]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
0	1	526301100	20	RL	141	31770	Pave	<NA>	
1	2	526350040	20	RH	80	11622	Pave	<NA>	
2	3	526351010	20	RL	81	14267	Pave	<NA>	
3	4	526353030	20	RL	93	11160	Pave	<NA>	
4	5	527105010	60	RL	74	13830	Pave	<NA>	
...	...	...	...	...	...	...	...	...	
2925	2926	923275080	80	RL	37	7937	Pave	<NA>	
2926	2927	923276100	20	RL	<NA>	8885	Pave	<NA>	
2927	2928	923400125	85	RL	62	10441	Pave	<NA>	
2928	2929	924100070	20	RL	77	10010	Pave	<NA>	
2929	2930	924151050	60	RL	74	9627	Pave	<NA>	

2930 rows × 82 columns

In [14]:

```
# Examining unique values
# Note the empty string
(df
 .Electrical
 .value_counts()
)
```

Out[14]:

```
Electrical
SBrkr    2682
FuseA     188
FuseF      50
FuseP       8
Mix         1
Name: count, dtype: int64[pyarrow]
```

In [15]:

```
# Converting to Category
(df
 .assign(
     **df
     .select_dtypes('string')
     .replace('', 'Not Applicable')
     .astype('category')
 )
)
```

Out[15]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
0	1	526301100	20	RL	141	31770	Pave	<NA>	
1	2	526350040	20	RH	80	11622	Pave	<NA>	
2	3	526351010	20	RL	81	14267	Pave	<NA>	
3	4	526353030	20	RL	93	11160	Pave	<NA>	
4	5	527105010	60	RL	74	13830	Pave	<NA>	
...	...	...	...	...	...	...	...	...	
2925	2926	923275080	80	RL	37	7937	Pave	<NA>	
2926	2927	923276100	20	RL	<NA>	8885	Pave	<NA>	
2927	2928	923400125	85	RL	62	10441	Pave	<NA>	
2928	2929	924100070	20	RL	77	10010	Pave	<NA>	
2929	2930	924151050	60	RL	74	9627	Pave	<NA>	

2930 rows × 82 columns

Cleaning Numbers

In [16]:

```
(df
 .select_dtypes(int)
 .describe()
)
```

Out[16]:

	Order	PID	MS SubClass	Lot Frontage	Lot Area	Ove Q
count	2930.0	2930.0	2930.0	2440.0	2930.0	293
mean	1465.5	714464496.988737	57.387372	69.22459	10147.921843	6.094
std	845.96247	188730844.64939	42.638025	23.365335	7880.017759	1.411
min	1.0	526301100.0	20.0	21.0	1300.0	
25%	733.25	528477022.5	20.0	58.0	7440.25	
50%	1465.5	535453620.0	50.0	68.0	9436.5	
75%	2197.75	907181097.5	70.0	80.0	11555.25	
max	2930.0	1007100110.0	190.0	313.0	215245.0	1

8 rows × 39 columns

In [17]:

```
(df
 .query('`Lot Frontage`.isna()')
)
```

Out[17]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
11	12	527165230	20	RL	<NA>	7980	Pave	<NA>	
14	15	527182190	120	RL	<NA>	6820	Pave	<NA>	
22	23	527368020	60	FV	<NA>	7500	Pave	<NA>	
23	24	527402200	20	RL	<NA>	11241	Pave	<NA>	
24	25	527402250	20	RL	<NA>	12537	Pave	<NA>	
...	...	...	...	...	...	...	...	...	
2894	2895	916326010	20	RL	<NA>	16669	Pave	<NA>	
2897	2898	916403130	60	RL	<NA>	11170	Pave	<NA>	
2898	2899	916460070	20	RL	<NA>	8098	Pave	<NA>	
2912	2913	923226150	90	RL	<NA>	11836	Pave	<NA>	
2926	2927	923276100	20	RL	<NA>	8885	Pave	<NA>	

490 rows × 82 columns

In [18]:

```
# How to see more data
with pd.option_context('display.min_rows', 30, 'display.max_columns', 82):
    display(df
     .query('`Lot Frontage`.isna()')
    )
```

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sha
11	12	527165230	20	RL	<NA>	7980	Pave	<NA>	
14	15	527182190	120	RL	<NA>	6820	Pave	<NA>	
22	23	527368020	60	FV	<NA>	7500	Pave	<NA>	I
23	24	527402200	20	RL	<NA>	11241	Pave	<NA>	
24	25	527402250	20	RL	<NA>	12537	Pave	<NA>	
55	56	528240070	60	RL	<NA>	7851	Pave	<NA>	I
57	58	528250100	80	RL	<NA>	7750	Pave	<NA>	
58	59	528292020	60	RL	<NA>	9505	Pave	<NA>	
74	75	531380080	60	RL	<NA>	8880	Pave	<NA>	
79	80	531452180	60	RL	<NA>	9453	Pave	<NA>	
86	87	532377060	20	RL	<NA>	9819	Pave	<NA>	
88	89	532378110	20	RL	<NA>	6897	Pave	<NA>	
99	100	533213030	20	FV	<NA>	4403	Pave	<NA>	
100	101	533221090	160	FV	<NA>	2117	Pave	<NA>	I
101	102	533221110	160	FV	<NA>	2980	Pave	<NA>	I
...	...	...	...	...	...	...	...	...	
2790	2791	907252050	60	RL	<NA>	9930	Pave	<NA>	
2792	2793	907255010	20	RL	<NA>	11088	Pave	<NA>	I
2793	2794	907255050	20	RL	<NA>	14781	Pave	<NA>	
2795	2796	907265030	20	RL	<NA>	8125	Pave	<NA>	I
2797	2798	907275030	60	RL	<NA>	21533	Pave	<NA>	
2845	2846	909131125	190	RH	<NA>	7082	Pave	<NA>	I
2859	2860	909276010	70	RL	<NA>	11435	Pave	<NA>	
2871	2872	909475020	20	RL	<NA>	16381	Pave	<NA>	
2892	2893	916252170	120	RM	<NA>	8239	Pave	<NA>	
2893	2894	916325040	20	RL	<NA>	50102	Pave	<NA>	
2894	2895	916326010	20	RL	<NA>	16669	Pave	<NA>	
2897	2898	916403130	60	RL	<NA>	11170	Pave	<NA>	
2898	2899	916460070	20	RL	<NA>	8098	Pave	<NA>	
2912	2913	923226150	90	RL	<NA>	11836	Pave	<NA>	
2926	2927	923276100	20	RL	<NA>	8885	Pave	<NA>	

490 rows × 82 columns

In [19]:
```
with pd.option_context('display.min_rows', 30, 'display.max_columns', 82):
    display(df
     .query('`Lot Frontage`.isna()')
     .style
     .set_sticky(axis='columns')
     .set_sticky(axis='index')
    )
```

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
11	12	527165230	20	RL		7980	Pave		
14	15	527182190	120	RL		6820	Pave		
22	23	527368020	60	FV		7500	Pave		
23	24	527402200	20	RL		11241	Pave		
24	25	527402250	20	RL		12537	Pave		
55	56	528240070	60	RL		7851	Pave		
57	58	528250100	80	RL		7750	Pave		
58	59	528292020	60	RL		9505	Pave		
74	75	531380080	60	RL		8880	Pave		
79	80	531452180	60	RL		9453	Pave		
86	87	532377060	20	RL		9819	Pave		
88	89	532378110	20	RL		6897	Pave		
99	100	533213030	20	FV		4403	Pave		
100	101	533221090	160	FV		2117	Pave		
101	102	533221110	160	FV		2980	Pave		
103	104	533223100	160	FV		2403	Pave		
108	109	533352170	60	RL		13517	Pave		
110	111	534129040	20	RL		10456	Pave		
112	113	534152050	20	RL		10603	Pave		
113	114	534152070	50	RL		18837	Pave		
118	119	534251320	20	RL		9790	Pave		
122	123	534403360	80	RL		10600	Pave	Pave	
123	124	534403410	80	RL		14112	Pave		
136	137	535125010	20	RL		19900	Pave		
140	141	535152130	20	RL		8050	Pave		
144	145	535154050	20	RL		12160	Pave		
159	160	535401080	20	RL		9830	Pave		
180	181	902206240	50	RM		8239	Pave		
192	193	903206120	75	RL		7793	Pave		
208	209	904100140	70	RL		24090	Pave		
213	214	904351040	70	C (all)		6449	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
219	220	905103060	20	RL		11341	Pave		
221	222	905105070	20	RL		8246	Pave		
225	226	905107110	90	RL		7424	Pave		
227	228	905107320	60	RL		11616	Pave		
229	230	905109170	20	RL		20062	Pave		
232	233	905325030	40	RL		23595	Pave		
233	234	905352140	60	RL		17082	Pave		
242	243	905475510	20	RL		11200	Pave		
257	258	907180050	60	RL		9337	Pave		
260	261	907200290	60	RL		10900	Pave		
264	265	907252120	20	RL		11423	Pave		
268	269	907290170	120	RM		4435	Pave		
269	270	907290240	120	RM		4426	Pave		
289	290	909176150	30	RL		7890	Pave		
312	313	914478045	80	RL		12328	Pave		
313	314	914478110	90	RL		12760	Pave		
314	315	916125360	20	RL		57200	Pave		
325	326	923205015	20	RL		11875	Pave		
326	327	923225300	160	RM		1974	Pave		
334	335	923251080	20	RL		26142	Pave		
345	346	527105130	60	RL		11792	Pave		
348	349	527110020	80	RL		8530	Pave		
357	358	527163070	60	RL		9765	Pave		
358	359	527163130	60	RL		8803	Pave		
360	361	527164060	60	RL		9636	Pave		
362	363	527165130	20	RL		9248	Pave		
363	364	527166010	60	RL		10762	Pave		
365	366	527182110	120	RL		5814	Pave		
373	374	527326150	20	RL		16635	Pave		
374	375	527352150	60	RL		13250	Pave		
376	377	527353060	60	RL		12388	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
377	378	527354100	80	RL		14115	Pave		
382	383	527359180	60	RL		10304	Pave		
385	386	527366030	60	FV		7500	Pave		
386	387	527368010	60	FV		8470	Pave		
387	388	527375100	20	RL		9373	Pave		
391	392	527378140	80	RL		10448	Pave		
393	394	527402220	20	RL		8750	Pave		
395	396	527404020	20	RL		7830	Pave		
396	397	527404030	20	RL		8510	Pave		
408	409	527452060	120	RL		4928	Pave		
419	420	527455280	20	RL		10710	Pave		
475	476	528235090	60	RL		8068	Pave		
478	479	528240060	80	RL		7750	Pave		
480	481	528250020	60	RL		8965	Pave		
481	482	528250040	60	RL		8174	Pave		
483	484	528275070	60	RL		8795	Pave		
484	485	528275160	60	RL		12891	Pave		
485	486	528280230	60	RL		12224	Pave		
490	491	528292030	60	RL		15896	Pave		
491	492	528292040	60	RL		24682	Pave		
492	493	528292070	60	RL		8755	Pave		
497	498	528344040	60	RL		16545	Pave		
500	501	528363050	20	RL		10750	Pave		
503	504	528387030	60	RL		11692	Pave		
505	506	528390210	60	RL		29959	Pave		
550	551	531453100	60	RL		10274	Pave		
556	557	532354160	20	RL		8499	Pave		
557	558	532354230	20	RL		9079	Pave		
558	559	532376070	20	RL		9316	Pave		
559	560	532376110	20	RL		7791	Pave		
563	564	532478020	20	RL		15676	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
564	565	533135020	60	RL		11949	Pave		
568	569	533213010	120	FV		3830	Pave	Pave	
569	570	533213020	120	FV		4217	Pave	Pave	
574	575	533252040	20	RL		14694	Pave		
578	579	533352150	20	RL		9991	Pave		
580	581	534127130	20	RL		11717	Pave		
581	582	534127170	20	RL		9156	Pave		
582	583	534128010	60	RL		10382	Pave		
583	584	534128020	60	RL		12732	Pave		
584	585	534128100	60	RL		12936	Pave		
586	587	534129080	80	RL		17871	Pave		
589	590	534151120	60	RL		13774	Pave		
597	598	534251030	85	RL		16500	Pave		
598	599	534252240	20	RL		9790	Pave		
602	603	534277090	20	RL		9450	Pave		
603	604	534278070	20	RL		13495	Pave		
608	609	534402140	20	RL		11000	Pave		
609	610	534402170	60	RL		8970	Pave		
610	611	534403370	80	RL		12095	Pave		
615	616	534451110	50	RL		7015	Pave		
624	625	535105100	20	RL		9500	Pave		
629	630	535150070	50	RL		12513	Pave		
632	633	535154060	20	RL		12285	Pave		
673	674	535425070	20	RL		17600	Pave		
691	692	902101010	50	RM		3950	Pave	Grvl	
720	721	902331010	30	C (all)		3300	Pave		
729	730	903201020	30	RL		6615	Pave		
732	733	903205040	30	RL		8854	Pave		
747	748	903400220	75	RL		11888	Pave	Pave	
756	757	903475100	70	RM		5775	Pave		
764	765	904301100	90	RL		10547	Pave		

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
765	766	904301375	30	RL		10020	Pave		
773	774	905107310	85	RL		8014	Pave		
774	775	905108190	85	RL		7252	Pave		
776	777	905200220	20	RL		11616	Pave		
779	780	905225090	80	RL		15584	Pave		
780	781	905228050	20	RL		9000	Pave		
781	782	905229040	50	RL		11250	Pave		
783	784	905377010	20	RL		17140	Pave		
784	785	905377130	30	RL		12342	Pave		
785	786	905401100	20	RL		10708	Pave		
786	787	905402060	20	RL		13680	Pave		
787	788	905402070	20	RL		15635	Pave		
794	795	905475500	20	RL		11500	Pave		
815	816	906230010	90	RL		11855	Pave		
816	817	906230020	90	RL		7939	Pave		
817	818	906230030	90	RL		7976	Pave		
832	833	906402060	80	RL		12800	Pave		
833	834	906426210	60	RL		16698	Pave		
834	835	906475070	60	RL		28698	Pave		
853	854	907201220	20	RL		16269	Pave		
856	857	907202080	20	RL		7000	Pave		
857	858	907202130	20	RL		9286	Pave		
863	864	907252060	60	RL		12334	Pave		
864	865	907252210	20	RL		11838	Pave		
865	866	907252220	60	RL		11885	Pave		
866	867	907253130	60	RL		11050	Pave		
868	869	907265010	60	RL		11250	Pave		
871	872	907275140	20	RL		12782	Pave		
873	874	907285020	60	RL		9375	Pave		
877	878	907290180	120	RM		4435	Pave		
878	879	907290210	120	RM		4435	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
901	902	908276150	20	RL		8926	Pave		
920	921	909276110	70	RL		7500	Pave		
925	926	909279010	90	RL		8145	Pave		
936	937	909452050	80	RL		13607	Pave		
937	938	909475040	20	RL		17597	Pave		
938	939	909475300	20	RL		21695	Pave		
953	954	914476380	80	RL		9947	Pave		
955	956	916176030	20	RL		14375	Pave		
963	964	916403200	60	RL		9839	Pave		
965	966	916455070	20	RL		6853	Pave		
972	973	923203190	120	RM		4500	Pave		
982	983	923275040	85	RL		9101	Pave		
983	984	923275140	20	RL		8780	Pave		
987	988	924100040	20	RL		9819	Pave		
990	991	526353050	20	RL		12925	Pave		
996	997	527107010	60	RL		15038	Pave		
1005	1006	527163100	60	RL		8000	Pave		
1006	1007	527164120	60	RL		10832	Pave		
1007	1008	527165010	60	RL		14067	Pave		
1013	1014	527226020	20	RL		31220	Pave		
1015	1016	527276040	20	RL		47280	Pave		
1019	1020	527302070	20	RL		10825	Pave		
1022	1023	527325070	60	RL		12227	Pave		
1024	1025	527326130	20	RL		15611	Pave		
1027	1028	527357180	60	RL		12511	Pave		
1032	1033	527380240	60	RL		14311	Pave		
1037	1038	527425035	20	RL		12735	Pave		
1049	1050	527455270	20	RL		9477	Pave		
1080	1081	528228345	120	RL		3940	Pave		
1081	1082	528228405	120	RM		3940	Pave		
1089	1090	528240050	60	RL		8010	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1090	1091	528250030	60	RL		8396	Pave		
1095	1096	528290090	60	RL		7750	Pave		
1101	1102	528326110	60	RL		11000	Pave		
1104	1105	528363020	60	RL		11929	Pave		
1145	1146	531450090	20	RL		7153	Pave		
1150	1151	532353050	20	RL		12968	Pave		
1157	1158	533125080	60	RL		9205	Pave		
1167	1168	533215020	120	FV		4765	Pave		
1168	1169	533215030	120	FV		4538	Pave		
1181	1182	533251120	20	RL		11120	Pave		
1182	1183	533350090	60	RL		24572	Pave		
1184	1185	534104100	60	FV		7500	Pave		
1186	1187	534127210	80	RL		11104	Pave		
1189	1190	534129060	20	RL		15387	Pave		
1198	1199	534250335	60	RL		13355	Pave		
1199	1200	534250370	60	RL		8963	Pave		
1201	1202	534251280	60	RL		9130	Pave		
1202	1203	534252090	85	RL		12122	Pave		
1203	1204	534252270	60	RL		9900	Pave		
1206	1207	534277070	20	RL		8475	Pave		
1217	1218	534428020	20	RL		12493	Pave		
1218	1219	534428100	20	RL		11332	Pave		
1219	1220	534451080	20	RL		6627	Pave		
1228	1229	535103050	60	RL		13700	Pave		
1234	1235	535150210	20	RL		7390	Pave		
1247	1248	535302140	20	RL		12774	Pave		
1263	1264	535426260	20	RL		10920	Pave		
1264	1265	535426350	20	RL		12929	Pave		
1319	1320	902401010	50	RM		5700	Pave		
1328	1329	903204010	50	RM		7425	Pave		
1330	1331	903206070	50	RL		7010	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1343	1344	903232030	30	RM		6120	Pave		
1354	1355	903401050	50	RL		9144	Pave	Pave	
1357	1358	903427090	70	RM		5100	Pave	Grvl	
1359	1360	903452025	30	RM		6291	Grvl		
1362	1363	903455030	50	RM		10320	Pave	Grvl	
1365	1366	903458110	50	RM		7920	Pave	Grvl	
1376	1377	905100020	85	RL		11235	Pave		
1379	1380	905104080	20	RL		7162	Pave		
1382	1383	905107280	85	RL		7703	Pave		
1383	1384	905107380	20	RL		9981	Pave		
1384	1385	905108170	85	RL		7400	Pave		
1387	1388	905200160	20	RL		9000	Pave		
1388	1389	905200510	20	RL		8544	Pave		
1391	1392	905201120	20	RL		13284	Pave		
1392	1393	905202230	20	RL		13500	Pave		
1395	1396	905226110	190	RL		10532	Pave		
1397	1398	905300020	80	RL		10200	Pave		
1398	1399	905351089	120	RL		2887	Pave		
1402	1403	905401060	20	RL		53227	Pave		
1419	1420	906204280	60	RL		9771	Pave		
1421	1422	906223140	60	RL		14171	Pave		
1428	1429	906424010	80	RL		11454	Pave		
1429	1430	906475100	20	RL		11500	Pave		
1434	1435	907175080	20	RL		8696	Pave		
1435	1436	907176010	60	RL		13142	Pave		
1444	1445	907200110	20	RL		9200	Pave		
1446	1447	907202010	20	RL		12250	Pave		
1447	1448	907202160	80	RL		10970	Pave		
1448	1449	907202190	20	RL		9216	Pave		
1455	1456	907253060	60	RL		10316	Pave		
1456	1457	907253110	60	RL		10400	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1459	1460	907255020	60	RL		9240	Pave		
1460	1461	907255030	60	RL		9720	Pave		
1461	1462	907255060	20	RL		14860	Pave		
1462	1463	907260010	60	RL		11250	Pave		
1469	1470	907290250	120	RM		4426	Pave		
1481	1482	907425010	120	RM		4426	Pave		
1482	1483	907425015	120	RM		4426	Pave		
1483	1484	907425030	120	RM		4438	Pave		
1484	1485	907425035	120	RM		4438	Pave		
1494	1495	908151040	80	RL		9638	Pave		
1516	1517	909131170	70	RH		12155	Pave		
1530	1531	909275040	70	RL		9650	Pave		
1533	1534	909277040	50	RL		11700	Pave	Grvl	
1534	1535	909277070	50	RL		9260	Pave	Grvl	
1537	1538	909282030	50	RL		14100	Pave		
1538	1539	909425010	50	RL		15660	Pave		
1541	1542	909428190	20	RL		14778	Pave		
1547	1548	910202100	30	RM		5890	Pave		
1563	1564	914453045	20	RL		23730	Pave		
1564	1565	914465060	20	RL		13265	Pave		
1565	1566	914467050	60	RL		11050	Pave		
1570	1571	916125425	190	RL		164660	Grvl		
1572	1573	916325080	20	RL		15498	Pave		
1584	1585	916460060	20	RL		7915	Pave		
1593	1594	923225080	120	RM		4224	Pave		
1594	1595	923225150	160	RM		2665	Pave		
1598	1599	923227030	20	RL		17979	Pave		
1609	1610	924100020	60	RL		11075	Pave		
1610	1611	1007100110	70	I (all)		56600	Pave		
1615	1616	527105140	60	RL		12394	Pave		
1616	1617	527107040	60	RL		10364	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1617	1618	527110080	60	RL		13869	Pave		
1621	1622	527158090	80	RL		10147	Pave		
1622	1623	527161090	60	RL		8637	Pave		
1626	1627	527163080	20	RL		9556	Pave		
1627	1628	527165020	80	RL		10784	Pave		
1628	1629	527165100	80	RL		9125	Pave		
1629	1630	527165170	60	RL		7655	Pave		
1631	1632	527182040	120	RL		3696	Pave		
1632	1633	527182170	160	RL		5062	Pave		
1643	1644	527301080	20	RL		12546	Pave		
1644	1645	527301280	20	RL		10960	Pave		
1650	1651	527327050	60	RL		12046	Pave		
1651	1652	527328020	80	RL		10395	Pave		
1659	1660	527359080	60	RL		12384	Pave		
1663	1664	527402150	20	RL		10530	Pave		
1664	1665	527402240	60	RL		7472	Pave		
1669	1670	527404150	20	RL		7340	Pave		
1670	1671	527425025	20	RL		17199	Pave		
1680	1681	527452100	120	RL		4928	Pave		
1711	1712	528172030	60	RL		12568	Pave		
1743	1744	528228375	120	RL		3621	Pave		
1751	1752	528250010	80	RL		11950	Pave		
1753	1754	528275035	60	RL		8063	Pave		
1754	1755	528275110	60	RL		8740	Pave		
1757	1758	528290060	60	RL		7750	Pave		
1762	1763	528326060	60	RL		11000	Pave		
1763	1764	528327060	20	RL		11400	Pave		
1772	1773	528366050	20	RL		12692	Pave		
1811	1812	531384070	60	RL		11613	Pave		
1813	1814	531385130	20	RL		16196	Pave		
1816	1817	531453140	85	RL		9180	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1821	1822	532354070	20	RL		7758	Pave		
1825	1826	532377140	20	RL		9945	Pave		
1826	1827	532378240	20	RL		6173	Pave		
1827	1828	532476080	60	RL		19522	Pave		
1828	1829	532477040	60	RL		17542	Pave		
1830	1831	532479120	85	RL		16647	Pave		
1832	1833	533120030	60	RL		9572	Pave		
1844	1845	533221030	160	FV		2117	Pave		
1846	1847	533221100	160	FV		2117	Pave		
1847	1848	533223050	160	FV		5105	Pave		
1851	1852	533242090	60	FV		8010	Pave	Pave	
1855	1856	533251130	80	RL		16157	Pave		
1861	1862	533352075	90	RL		18890	Pave		
1862	1863	534104090	60	FV		7050	Pave		
1865	1866	534127140	85	RL		8723	Pave		
1870	1871	534175010	90	RL		11500	Pave		
1875	1876	534202030	20	RL		10355	Pave		
1877	1878	534252040	20	RL		9503	Pave		
1878	1879	534252060	90	RL		10624	Pave		
1879	1880	534252070	90	RL		10899	Pave		
1880	1881	534252110	20	RL		12342	Pave		
1881	1882	534275170	20	RL		12772	Pave		
1886	1887	534276290	20	RL		8339	Pave		
1887	1888	534278150	20	RL		14357	Pave		
1894	1895	534403420	20	RL		11382	Pave		
1895	1896	534425015	20	RL		22002	Pave		
1896	1897	534425080	20	RL		14585	Pave		
1911	1912	535102010	85	RL		10050	Pave		
1915	1916	535126180	60	RL		18450	Pave		
1927	1928	535181030	20	RL		12155	Pave		
1941	1942	535353130	20	RL		15783	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1944	1945	535354260	50	RL		12099	Pave		
1963	1964	535453080	20	RL		7500	Pave		
1989	1990	902300020	70	RM		10337	Pave	Pave	
2013	2014	903231090	50	RM		6240	Pave		
2026	2027	903426010	70	RM		5700	Pave		
2030	2031	903450060	50	RM		7758	Pave		
2040	2041	903475040	50	RM		12358	Pave		
2053	2054	905103110	20	RL		11677	Pave		
2056	2057	905104170	20	RL		8978	Pave		
2059	2060	905105170	20	RL		8398	Pave		
2063	2064	905200010	20	RL		8169	Pave		
2066	2067	905226050	30	RL		25339	Pave		
2070	2071	905228020	20	RL		9000	Pave		
2071	2072	905301050	20	RL		115149	Pave		
2072	2073	905352010	20	RL		11075	Pave		
2074	2075	905376090	20	RL		17541	Pave		
2075	2076	905377020	20	RL		22692	Pave		
2081	2082	905475520	30	RL		11515	Pave		
2113	2114	906402070	60	RL		14364	Pave		
2114	2115	906403060	60	RL		8883	Pave		
2115	2116	906426060	50	RL		159000	Pave		
2116	2117	906426195	60	RL		53107	Pave		
2117	2118	906475110	60	RL		12205	Pave		
2119	2120	907125040	20	RL		14217	Pave		
2123	2124	907131120	60	RL		9531	Pave		
2131	2132	907192040	60	RL		8826	Pave		
2145	2146	907252020	60	RL		9375	Pave		
2146	2147	907252190	20	RL		11354	Pave		
2152	2153	907275150	60	RL		12728	Pave		
2153	2154	907280090	60	RL		15295	Pave		
2175	2176	908152070	20	RL		7917	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
2176	2177	908152180	90	RL		9555	Pave		
2216	2217	909279080	50	RL		11275	Pave		
2222	2223	909428120	20	RL		21000	Pave		
2223	2224	909428180	20	RL		25485	Pave		
2224	2225	909428340	20	RL		21579	Pave		
2226	2227	909452102	20	RL		17871	Pave		
2228	2229	909475050	20	RL		20693	Pave		
2229	2230	909475070	20	RL		32668	Pave		
2247	2248	914452090	85	RL		12150	Pave		
2248	2249	914452120	85	RL		7540	Pave		
2252	2253	914476330	20	RL		9928	Pave		
2253	2254	914478020	80	RL		8750	Pave		
2256	2257	916253320	120	RM		9763	Pave		
2266	2267	916455010	60	RL		9303	Pave		
2267	2268	916455050	20	RL		6718	Pave		
2270	2271	916460020	20	RL		7777	Pave		
2272	2273	916477060	60	RL		11800	Pave		
2282	2283	923205025	190	RL		32463	Pave		
2286	2287	923228200	180	RM		1533	Pave		
2291	2292	923229010	80	RL		11333	Pave		
2293	2294	923229100	80	RL		15957	Pave		
2300	2301	923275010	20	RL		11000	Pave		
2305	2306	526302030	20	RL		11027	Pave		
2307	2308	526302120	20	RL		11765	Pave		
2308	2309	526303060	20	RL		39384	Pave		
2311	2312	527106010	60	RL		13006	Pave		
2312	2313	527107020	60	RL		13041	Pave		
2313	2314	527107030	60	RL		13031	Pave		
2323	2324	527158020	20	RL		8076	Pave		
2324	2325	527163020	60	RL		7685	Pave		
2328	2329	527190220	120	RL		6563	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
2337	2338	527226010	60	RL		14762	Pave		
2343	2344	527302080	50	RL		13837	Pave		
2344	2345	527325160	60	RL		16659	Pave		
2345	2346	527327080	60	RL		18800	Pave		
2346	2347	527328010	85	RL		10464	Pave		
2352	2353	527358090	85	RL		9927	Pave		
2359	2360	527402210	20	RL		15870	Pave		
2361	2362	527403120	20	RL		8125	Pave		
2402	2403	528172150	60	RL		13215	Pave		
2409	2410	528188040	120	RL		3136	Pave		
2420	2421	528228325	120	RL		3196	Pave		
2421	2422	528228340	120	RL		3196	Pave		
2422	2423	528228360	120	RL		2938	Pave		
2423	2424	528228415	120	RM		3072	Pave		
2424	2425	528228430	120	RM		3072	Pave		
2431	2432	528235050	60	RL		7861	Pave		
2436	2437	528275060	60	RL		8121	Pave		
2437	2438	528275080	60	RL		8658	Pave		
2438	2439	528280100	60	RL		11214	Pave		
2440	2441	528292080	60	RL		12104	Pave		
2446	2447	528327010	60	RL		9233	Pave		
2451	2452	528363070	60	RL		10236	Pave		
2453	2454	528366040	60	RL		12585	Pave		
2482	2483	531452210	60	RL		9019	Pave		
2483	2484	531478010	20	RH		8900	Pave		
2488	2489	532353030	20	RL		9240	Pave		
2490	2491	532376080	20	RL		9308	Pave		
2492	2493	532376250	20	RL		8638	Pave		
2493	2494	532378050	20	RL		13052	Pave		
2494	2495	532378070	20	RL		13526	Pave		
2495	2496	532378130	20	RL		8020	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
2497	2498	532378220	20	RL		8789	Pave		
2501	2502	533127080	60	RL		14541	Pave		
2502	2503	533128030	60	RL		13346	Pave		
2510	2511	533221080	160	FV		2998	Pave		
2512	2513	533223080	160	FV		2651	Pave		
2513	2514	533223110	160	FV		4447	Pave		
2517	2518	533252020	20	RL		11250	Pave		
2518	2519	533253030	120	RL		3760	Pave		
2525	2526	534127190	20	RL		20781	Pave		
2527	2528	534128210	60	RL		11029	Pave		
2537	2538	534202020	20	RL		9759	Pave		
2540	2541	534250300	60	RL		14803	Pave		
2541	2542	534275010	20	RL		10659	Pave		
2546	2547	534403400	20	RL		10368	Pave		
2552	2553	534430110	20	RL		11425	Pave		
2565	2566	535101110	90	RL		8917	Pave		
2566	2567	535103070	80	RL		12700	Pave		
2576	2577	535177100	20	RL		9610	Pave		
2581	2582	535301010	90	RL		7032	Pave		
2617	2618	535425080	60	RL		18275	Pave		
2625	2626	535454050	90	RL		8544	Pave		
2670	2671	903200050	30	RL		7446	Pave		
2676	2677	903231190	50	RM		6240	Pave		
2708	2709	905103130	20	RL		11327	Pave		
2712	2713	905106210	20	RL		11553	Pave		
2714	2715	905107220	20	RL		9535	Pave		
2716	2717	905107300	80	RL		7176	Pave		
2717	2718	905108090	90	RL		9662	Pave		
2719	2720	905200280	50	RL		13650	Pave		
2722	2723	905200380	30	RL		17529	Pave		
2723	2724	905200490	80	RL		10246	Pave		

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
2724	2725	905201090	20	RL		14175	Pave		
2725	2726	905202190	20	RL		20355	Pave		
2730	2731	905351045	150	RL		1700	Pave		
2731	2732	905351150	120	RL		5271	Pave		
2735	2736	905426150	80	RL		19690	Pave		
2746	2747	906202040	20	RL		11200	Pave		
2764	2765	906426090	20	RL		36500	Pave		
2765	2766	906475050	80	RL		21453	Pave		
2771	2772	907131070	20	RL		8685	Pave		
2788	2789	907230240	160	RH		3612	Pave		
2790	2791	907252050	60	RL		9930	Pave		
2792	2793	907255010	20	RL		11088	Pave		
2793	2794	907255050	20	RL		14781	Pave		
2795	2796	907265030	20	RL		8125	Pave		
2797	2798	907275030	60	RL		21533	Pave		
2845	2846	909131125	190	RH		7082	Pave		
2859	2860	909276010	70	RL		11435	Pave		
2871	2872	909475020	20	RL		16381	Pave		
2892	2893	916252170	120	RM		8239	Pave		
2893	2894	916325040	20	RL		50102	Pave		
2894	2895	916326010	20	RL		16669	Pave		
2897	2898	916403130	60	RL		11170	Pave		
2898	2899	916460070	20	RL		8098	Pave		
2912	2913	923226150	90	RL		11836	Pave		
2926	2927	923276100	20	RL		8885	Pave		

In [20]:

```
# Examine a column with missing values
(df
 .query('`Garage Yr Blt`.isna()')
 )
```

Out[20]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
27	28	527425090	20	RL	70	10500	Pave	<NA>	
119	120	534276360	20	RL	77	9320	Pave	<NA>	
125	126	534427010	90	RL	98	13260	Pave	<NA>	
129	130	534450180	20	RL	50	7207	Pave	<NA>	
130	131	534451150	30	RL	55	5350	Pave	<NA>	
...	...	...	...	...	...	...	...	...	
2913	2914	923226180	180	RM	21	1470	Pave	<NA>	
2916	2917	923228130	180	RM	21	1533	Pave	<NA>	
2918	2919	923228210	160	RM	21	1526	Pave	<NA>	
2919	2920	923228260	160	RM	21	1936	Pave	<NA>	
2927	2928	923400125	85	RL	62	10441	Pave	<NA>	

159 rows × 82 columns

In [21]:

```
# missing + 2207!!!?
(df
 ['Garage Yr Blt']
 .describe()
)
```

Out[21]:

```
count         2771.0
mean     1978.132443
std        25.528411
min         1895.0
25%         1960.0
50%         1979.0
75%         2002.0
max         2207.0
Name: Garage Yr Blt, dtype: double[pyarrow]
```

In [22]:

```
# probably a typo!!
with pd.option_context('display.min_rows', 30, 'display.max_columns', 82):
    display(df.query('`Garage Yr Blt` > 2200'))
```

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	L Sha
2260	2261	916384070	20	RL	68	8298	Pave	<NA>	II

In [23]:

```
# Any columns with Yr

(df
```

```
 .filter(like='Yr')
)
```

Out[23]:

	Garage Yr Blt	Yr Sold
0	1960	2010
1	1961	2010
2	1958	2010
3	1968	2010
4	1997	2010
...	...	...
2925	1984	2006
2926	1983	2006
2927	<NA>	2006
2928	1975	2006
2929	1993	2006

2930 rows × 2 columns

In [24]:

```
# Any columns with Yr > 2023
(df
 .filter(like='Yr')
 .pipe(lambda df_: df_[df_.gt(2023).any(axis='columns')])
)
```

Out[24]:

	Garage Yr Blt	Yr Sold
2260	2207	2007

In [25]:

```
# What about "Year" columns?
(df
 .rename(columns=lambda name: name.replace('Yr', 'Year'))
 .filter(like='Year')
 .pipe(lambda df_: df_[df_.gt(2023).any(axis='columns')])
)
```

Out[25]:

	Year Built	Year Remod/Add	Garage Year Blt	Year Sold
2260	2006	2007	2207	2007

In [26]:

```
# Garage Yr Blt -> clip to max of Year Built
(df
 ['Garage Yr Blt']
 .clip(upper=df['Year Built'].max())
 .value_counts()
 .sort_index()
)
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

Out[26]:
```
Garage Yr Blt
1895      1
1896      1
1900      6
1906      1
1908      1
       ...
2006    115
2007    115
2008     61
2009     29
2010      6
Name: count, Length: 102, dtype: int64[pyarrow]
```

In [27]:
```
df['Year Built'].max()
```

Out[27]:
```
2010
```

In [28]:
```
with pd.option_context('display.min_rows', 30, 'display.max_columns', 82):
    display(df
     .query('`Year Built`.max()')
    )
```

```
Order                   2011
PID                903227140
MS SubClass               70
MS Zoning                 RM
Lot Frontage              50
Lot Area                6000
Street                  Pave
Alley                   <NA>
Lot Shape                Reg
Land Contour             Lvl
Utilities             AllPub
Lot Config            Inside
Land Slope               Gtl
Neighborhood         BrkSide
Condition 1             Norm
                       ...
Wood Deck SF               0
Open Porch SF              0
Enclosed Porch             0
3Ssn Porch                 0
Screen Porch               0
Pool Area                  0
Pool QC                 <NA>
Fence                   GdWo
Misc Feature            <NA>
Misc Val                   0
Mo Sold                    2
Yr Sold                 2007
Sale Type                 WD
Sale Condition        Normal
SalePrice             128000
Name: 2010, Length: 82, dtype: object
```

In [29]:
```
# Update categories and clip
(df
 .assign(**df.select_dtypes('string').replace('', 'Missing').astype('categor
        **{'Garage Yr Blt': df['Garage Yr Blt'].clip(upper=df['Year Built']
 .dtypes.value_counts()
)
```

Out[29]:
```
int64[pyarrow]    39
category           8
category           2
category           2
category           2
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
category           1
Name: count, dtype: int64
```

Shrinking Numbers

In [30]:
```
def shrink_ints(df):
    mapping = {}
    for col in df.dtypes[df.dtypes=='int64[pyarrow]'].index:
        max_ = df[col].max()
        min_ = df[col].min()
        if min_ < 0:
            continue
        if max_ < 255:
            mapping[col] = 'uint8[pyarrow]'
```

```
        elif max_ < 65_535:
            mapping[col] = 'uint16[pyarrow]'
        elif max_ <  4294967295:
            mapping[col] = 'uint32[pyarrow]'
    return df.astype(mapping)

memory_after_shirkining = (df
 .assign(**df.select_dtypes('string').replace('', 'Missing').astype('categor
        **{'Garage Yr Blt': df['Garage Yr Blt'].clip(upper=df['Year Built']
 .pipe(shrink_ints)
 .memory_usage(deep=True)
 .sum()
)

memory= (df
 .memory_usage(deep=True)
 .sum()
)

print(f"Memory {memory/1000}mb \nAfter Shirking numbers-Memory is {memory_af
```

```
Memory 1847.796mb
After Shirking numbers-Memory is 360.288mb.
```

In [31]:

```
# make function and use pipe to join it
def shrink_ints(df):
    mapping = {}
    for col in df.dtypes[df.dtypes=='int64[pyarrow]'].index:
        max_ = df[col].max()
        min_ = df[col].min()
        if min_ < 0:
            continue
        if max_ < 255:
            mapping[col] = 'uint8[pyarrow]'
        elif max_ < 65_535:
            mapping[col] = 'uint16[pyarrow]'
        elif max_ <  4294967295:
            mapping[col] = 'uint32[pyarrow]'
    return df.astype(mapping)


def clean_housing_data(df):
    return (df
     .assign(**df.select_dtypes('string').replace('', 'Missing').astype('cat
            **{'Garage Yr Blt': df['Garage Yr Blt'].clip(upper=df['Year Bui
     .pipe(shrink_ints)
    )

clean_housing_data(df).dtypes
```

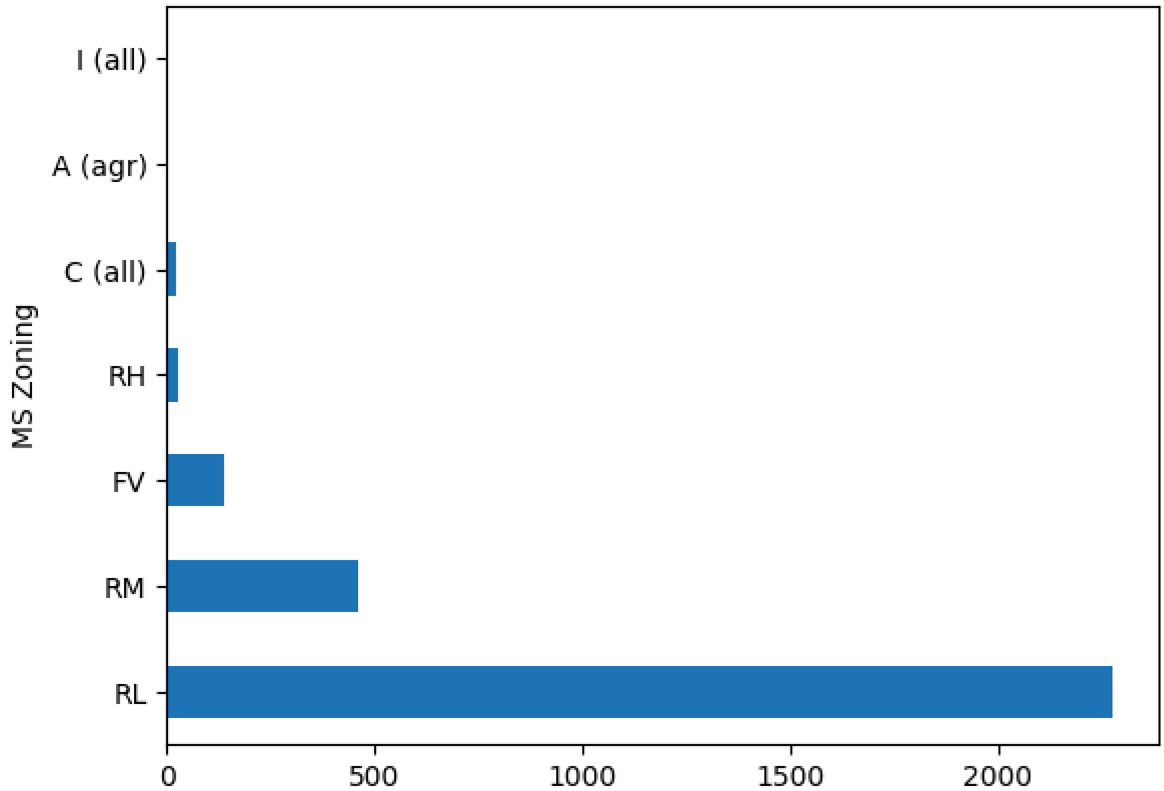
```
Out[31]: Order                uint16[pyarrow]
         PID                  uint32[pyarrow]
         MS SubClass           uint8[pyarrow]
         MS Zoning                   category
         Lot Frontage         uint16[pyarrow]
                                   ...
         Mo Sold               uint8[pyarrow]
         Yr Sold              uint16[pyarrow]
         Sale Type                   category
         Sale Condition              category
         SalePrice            uint32[pyarrow]
         Length: 82, dtype: object
```

Categorical Exploration

```
In [32]: import pandas as pd
url = 'housing_sale_data.csv'
raw = pd.read_csv(url, engine='pyarrow', dtype_backend='pyarrow')

# make function
def shrink_ints(df):
    mapping = {}
    for col in df.dtypes[df.dtypes=='int64[pyarrow]'].index:
        max_ = df[col].max()
        min_ = df[col].min()
        if min_ < 0:
            continue
        if max_ < 255:
            mapping[col] = 'uint8[pyarrow]'
        elif max_ < 65_535:
            mapping[col] = 'uint16[pyarrow]'
        elif max_ <  4294967295:
            mapping[col] = 'uint32[pyarrow]'
    return df.astype(mapping)


def clean_housing(df):
    return (df
     .assign(**df.select_dtypes('string').replace('', 'Missing').astype('cat
             **{'Garage Yr Blt': df['Garage Yr Blt'].clip(upper=df['Year Bui
     .pipe(shrink_ints)
    )

housing = clean_housing(raw)
```

```
In [33]: # categoricals
(housing
  ['MS Zoning']
  .value_counts()
  .plot.barh())
```

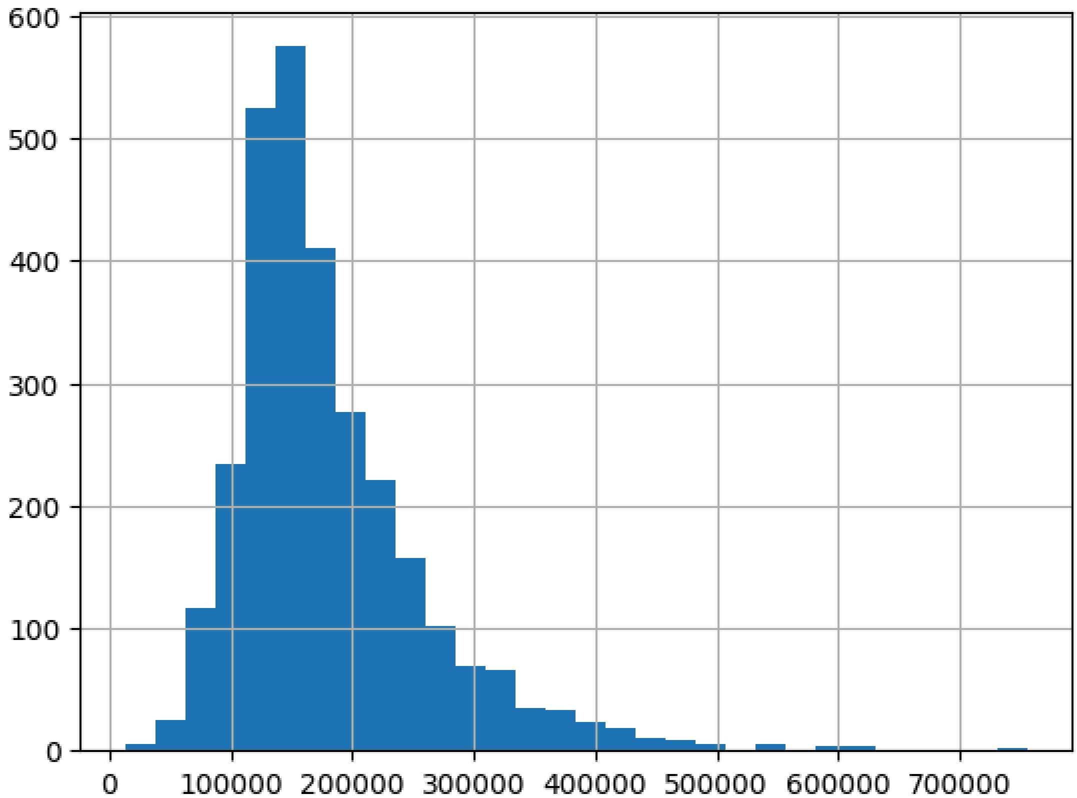
```
Out[33]: <Axes: ylabel='MS Zoning'>
```



Histograms and Distributions

In [34]:
```
# Numerical
(housing
 .SalePrice
 .hist(bins=30)
)
```

Out[34]: <Axes: >



Outliers and Z-scores

In [35]:
```
# outlier with Z-score
def calc_z(df, col):
    mean = df[col].mean()
    std = df[col].std()
    return (df[col]-mean)/std

(housing
 .pipe(calc_z, col='SalePrice')
)
```

Out[35]:
```
0       0.428156
1      -0.948795
2      -0.110107
3        0.79117
4       0.113961
          ...
2925    -0.47938
2926   -0.623334
2927   -0.610816
2928   -0.135142
2929    0.090177
Name: SalePrice, Length: 2930, dtype: double[pyarrow]
```

In [36]:
```
(housing
 .assign(z_score=calc_z(housing, col='SalePrice'))
 .query('z_score.abs() >= 3' or 'z_score <= -3')
)
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

Out[36]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
15	16	527216070	60	RL	47	53504	Pave	<NA>	
44	45	528150070	20	RL	100	12919	Pave	<NA>	
46	47	528176010	20	RL	110	14300	Pave	<NA>	
366	367	527214050	20	RL	63	17423	Pave	<NA>	
421	422	528102140	60	RL	110	14257	Pave	<NA>	
422	423	528104070	60	RL	104	13518	Pave	<NA>	
423	424	528106020	20	RL	105	15431	Pave	<NA>	
431	432	528110010	60	RL	97	13478	Pave	<NA>	
432	433	528110020	20	RL	105	13693	Pave	<NA>	
433	434	528110090	60	RL	107	13891	Pave	<NA>	
448	449	528166090	20	RL	110	15274	Pave	<NA>	
456	457	528176030	20	RL	100	14836	Pave	<NA>	
513	514	528441090	20	RL	85	11128	Pave	<NA>	
968	969	921128050	20	RL	85	12633	Pave	<NA>	
1051	1052	528102110	60	RL	96	12474	Pave	<NA>	
1053	1054	528104080	60	RL	67	14948	Pave	<NA>	
1059	1060	528118090	60	RL	96	12539	Pave	<NA>	
1063	1064	528164060	20	RL	106	12720	Pave	<NA>	
1067	1068	528178070	60	RL	130	16900	Pave	<NA>	
1425	1426	906412010	20	RL	91	11778	Pave	<NA>	
1637	1638	527216080	20	RL	52	51974	Pave	<NA>	
1641	1642	527256030	20	RL	85	14082	Pave	<NA>	
1642	1643	527256040	20	RL	81	13870	Pave	<NA>	
1691	1692	528106050	60	RL	107	13641	Pave	<NA>	
1693	1694	528106110	20	RL	105	15431	Pave	<NA>	
1695	1696	528110040	20	RL	107	13891	Pave	<NA>	
1699	1700	528114050	20	RL	110	14977	Pave	<NA>	
1701	1702	528118050	20	RL	59	17169	Pave	<NA>	
1760	1761	528320050	60	RL	160	15623	Pave	<NA>	
1763	1764	528327060	20	RL	<NA>	11400	Pave	<NA>	
1767	1768	528351010	60	RL	104	21535	Pave	<NA>	
1772	1773	528366050	20	RL	<NA>	12692	Pave	<NA>	

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
2097	2098	906340090	60	RL	77	9965	Pave	<NA>	
2329	2330	527210030	60	RL	59	16023	Pave	<NA>	
2330	2331	527210040	60	RL	60	18062	Pave	<NA>	
2332	2333	527212030	60	RL	85	16056	Pave	<NA>	
2334	2335	527214060	60	RL	82	16052	Pave	<NA>	
2336	2337	527216050	60	RL	66	13682	Pave	<NA>	
2341	2342	527256120	20	RL	90	18261	Pave	<NA>	
2382	2383	528110050	20	RL	107	13891	Pave	<NA>	
2400	2401	528170040	60	RL	56	20431	Pave	<NA>	
2445	2446	528320060	60	RL	118	35760	Pave	<NA>	
2450	2451	528360050	60	RL	114	17242	Pave	<NA>	
2456	2457	528429120	20	RL	49	20896	Pave	<NA>	
2666	2667	902400110	75	RM	90	22950	Pave	<NA>	

45 rows × 83 columns

In [37]:

```
def calc_iqr_outlier(df, col):
    ser = df[col]
    iqr = ser.quantile(.75) - ser.quantile(.25)
    med = ser.median()
    small_mask = ser < med-iqr*3
    large_mask = ser > med+iqr*3
    return small_mask | large_mask

(housing
 .assign(iqr_outlier=calc_iqr_outlier(housing, col='SalePrice'))
 .query('iqr_outlier')
)
```

Out[37]:

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
15	16	527216070	60	RL	47	53504	Pave	<NA>	
44	45	528150070	20	RL	100	12919	Pave	<NA>	
46	47	528176010	20	RL	110	14300	Pave	<NA>	
366	367	527214050	20	RL	63	17423	Pave	<NA>	
421	422	528102140	60	RL	110	14257	Pave	<NA>	
422	423	528104070	60	RL	104	13518	Pave	<NA>	
423	424	528106020	20	RL	105	15431	Pave	<NA>	
431	432	528110010	60	RL	97	13478	Pave	<NA>	
432	433	528110020	20	RL	105	13693	Pave	<NA>	
433	434	528110090	60	RL	107	13891	Pave	<NA>	
448	449	528166090	20	RL	110	15274	Pave	<NA>	
456	457	528176030	20	RL	100	14836	Pave	<NA>	
513	514	528441090	20	RL	85	11128	Pave	<NA>	
968	969	921128050	20	RL	85	12633	Pave	<NA>	
1051	1052	528102110	60	RL	96	12474	Pave	<NA>	
1053	1054	528104080	60	RL	67	14948	Pave	<NA>	
1056	1057	528110110	20	RL	105	13693	Pave	<NA>	
1059	1060	528118090	60	RL	96	12539	Pave	<NA>	
1063	1064	528164060	20	RL	106	12720	Pave	<NA>	
1064	1065	528166120	60	RL	110	13688	Pave	<NA>	
1067	1068	528178070	60	RL	130	16900	Pave	<NA>	
1425	1426	906412010	20	RL	91	11778	Pave	<NA>	
1637	1638	527216080	20	RL	52	51974	Pave	<NA>	
1641	1642	527256030	20	RL	85	14082	Pave	<NA>	
1642	1643	527256040	20	RL	81	13870	Pave	<NA>	
1690	1691	528106040	20	RL	107	14450	Pave	<NA>	
1691	1692	528106050	60	RL	107	13641	Pave	<NA>	
1693	1694	528106110	20	RL	105	15431	Pave	<NA>	
1695	1696	528110040	20	RL	107	13891	Pave	<NA>	
1699	1700	528114050	20	RL	110	14977	Pave	<NA>	
1700	1701	528118040	60	RL	118	13654	Pave	<NA>	
1701	1702	528118050	20	RL	59	17169	Pave	<NA>	

	Order	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Sl
1760	1761	528320050	60	RL	160	15623	Pave	<NA>	
1763	1764	528327060	20	RL	<NA>	11400	Pave	<NA>	
1767	1768	528351010	60	RL	104	21535	Pave	<NA>	
1772	1773	528366050	20	RL	<NA>	12692	Pave	<NA>	
1780	1781	528431040	20	RL	98	12291	Pave	<NA>	
2097	2098	906340090	60	RL	77	9965	Pave	<NA>	
2329	2330	527210030	60	RL	59	16023	Pave	<NA>	
2330	2331	527210040	60	RL	60	18062	Pave	<NA>	
2332	2333	527212030	60	RL	85	16056	Pave	<NA>	
2334	2335	527214060	60	RL	82	16052	Pave	<NA>	
2336	2337	527216050	60	RL	66	13682	Pave	<NA>	
2341	2342	527256120	20	RL	90	18261	Pave	<NA>	
2379	2380	528102080	60	RL	72	16387	Pave	<NA>	
2382	2383	528110050	20	RL	107	13891	Pave	<NA>	
2384	2385	528114010	20	RL	120	14780	Pave	<NA>	
2400	2401	528170040	60	RL	56	20431	Pave	<NA>	
2445	2446	528320060	60	RL	118	35760	Pave	<NA>	
2450	2451	528360050	60	RL	114	17242	Pave	<NA>	
2456	2457	528429120	20	RL	49	20896	Pave	<NA>	
2666	2667	902400110	75	RM	90	22950	Pave	<NA>	
2737	2738	905427030	75	RL	60	19800	Pave	<NA>	

53 rows × 83 columns

Correlations

In [38]:

```python
# Pearson correlation
housing.corr(numeric_only=True)
```

Out[38]:

	Order	PID	MS SubClass	Lot Frontage	Lot Area	Overall Qual	
Order	1.000000	0.173593	0.011797	-0.007034	0.031354	-0.048500	-0.
PID	0.173593	1.000000	-0.001281	-0.096918	0.034868	-0.263147	0.
MS SubClass	0.011797	-0.001281	1.000000	-0.420135	-0.204613	0.039419	-0.
Lot Frontage	-0.007034	-0.096918	-0.420135	1.000000	0.491313	0.212042	-0.
Lot Area	0.031354	0.034868	-0.204613	0.491313	1.000000	0.097188	-0.
Overall Qual	-0.048500	-0.263147	0.039419	0.212042	0.097188	1.000000	-0.
Overall Cond	-0.011054	0.104451	-0.067349	-0.074448	-0.034759	-0.094812	1.
Year Built	-0.052319	-0.343388	0.036579	0.121562	0.023258	0.597027	-0.
Year Remod/Add	-0.075566	-0.157111	0.043397	0.091712	0.021682	0.569609	0.
Mas Vnr Area	-0.030907	-0.229283	0.002730	0.222407	0.126830	0.429418	-0.
BsmtFin SF 1	-0.032321	-0.098375	-0.060075	0.215583	0.191555	0.284118	-0.
BsmtFin SF 2	-0.002773	-0.001145	-0.070946	0.045999	0.083150	-0.041287	0.
Bsmt Unf SF	0.005780	-0.087707	-0.130421	0.116743	0.023658	0.270058	-0.
Total Bsmt SF	-0.028719	-0.189642	-0.219445	0.353773	0.253589	0.547294	-0.
1st Flr SF	-0.013201	-0.141902	-0.247828	0.457391	0.332235	0.477837	-0.
2nd Flr SF	-0.000417	-0.003289	0.304237	0.029187	0.032996	0.241402	0.
Low Qual Fin SF	0.013589	0.056940	0.025765	0.005249	0.000812	-0.048680	0.
Gr Liv Area	-0.009342	-0.107579	0.068061	0.383822	0.285599	0.570556	-0.
Bsmt Full Bath	-0.042539	-0.037759	0.013701	0.108915	0.125877	0.167858	-0.
Bsmt Half Bath	0.024978	0.004328	-0.003329	-0.024724	0.026903	-0.041647	0.
Full Bath	-0.044985	-0.171431	0.134631	0.184521	0.127433	0.522263	-0.
Half Bath	-0.039749	-0.166636	0.175879	0.041880	0.035497	0.268853	-0.
Bedroom AbvGr	0.015424	0.006345	-0.019208	0.240442	0.136569	0.063291	-0.
Kitchen AbvGr	-0.017685	0.076470	0.257698	0.005407	-0.020301	-0.159744	-0.

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

	Order	PID	MS SubClass	Lot Frontage	Lot Area	Overall Qual	(
TotRms AbvGrd	0.002612	-0.068981	0.031898	0.353137	0.216597	0.380693	-0.
Fireplaces	-0.019156	-0.108056	-0.049955	0.257255	0.256989	0.393007	-0.
Garage Yr Blt	-0.054580	-0.263692	0.092526	0.077801	-0.008383	0.575140	-0.
Garage Cars	-0.036185	-0.237484	-0.045883	0.308706	0.179512	0.599545	-0.
Garage Area	-0.035435	-0.210606	-0.103239	0.358505	0.212822	0.563503	-0.
Wood Deck SF	-0.011292	-0.051135	-0.017310	0.120084	0.157212	0.255663	0.
Open Porch SF	0.016355	-0.071311	-0.014823	0.163040	0.103760	0.298412	-0.
Enclosed Porch	0.027908	0.162519	-0.022866	0.012758	0.021868	-0.140332	0.
3Ssn Porch	-0.024975	-0.024894	-0.037956	0.028564	0.016243	0.018240	0.
Screen Porch	0.004307	-0.025735	-0.050614	0.076666	0.055044	0.041615	0.
Pool Area	0.052518	-0.002845	-0.003434	0.173947	0.093775	0.030399	-0.
Misc Val	-0.006083	-0.008260	-0.029254	0.044476	0.069188	0.005179	0.
Mo Sold	0.133365	-0.050455	0.000350	0.011085	0.003859	0.031103	-0.
Yr Sold	-0.975993	0.009579	-0.017905	-0.007547	-0.023085	-0.020719	0.
SalePrice	-0.031408	-0.246521	-0.085092	0.357318	0.266549	0.799262	-0.

39 rows × 39 columns

In [39]:
```
(housing
 .corr(method='spearman', numeric_only=True)
 .style
 .background_gradient(cmap='RdBu', vmin=-1, vmax=1)
)
```

Out[39]:

	Order	PID	MS SubClass	Lot Frontage	Lot Area	Overall Qual	
Order	1.000000	0.205863	0.015074	-0.024209	0.012684	-0.049175	-0.
PID	0.205863	1.000000	-0.026875	-0.096820	-0.040342	-0.314353	0.
MS SubClass	0.015074	-0.026875	1.000000	-0.363408	-0.320550	0.103475	-0.
Lot Frontage	-0.024209	-0.096820	-0.363408	1.000000	0.659412	0.223162	-0.
Lot Area	0.012684	-0.040342	-0.320550	0.659412	1.000000	0.196855	-0.
Overall Qual	-0.049175	-0.314353	0.103475	0.223162	0.196855	1.000000	-0.
Overall Cond	-0.015534	0.111922	-0.065550	-0.104785	-0.079006	-0.189638	1.
Year Built	-0.056978	-0.314979	0.035632	0.192888	0.121151	0.664590	-0.
Year Remod/Add	-0.084144	-0.208414	0.015590	0.134932	0.103266	0.579323	-0.
Mas Vnr Area	-0.041604	-0.237169	-0.009247	0.275334	0.205822	0.423202	-0.
BsmtFin SF 1	-0.034229	-0.050349	-0.098222	0.156582	0.171376	0.179239	-0.
BsmtFin SF 2	-0.018105	0.012860	-0.093004	0.044513	0.057461	-0.091898	0.
Bsmt Unf SF	0.007011	-0.131413	-0.113501	0.092700	0.068211	0.239273	-0.
Total Bsmt SF	-0.023661	-0.188111	-0.300856	0.375535	0.352739	0.472852	-0.
1st Flr SF	-0.007155	-0.130427	-0.278139	0.442289	0.439129	0.415988	-0.
2nd Flr SF	-0.003142	-0.069092	0.478281	0.000692	0.064565	0.237677	-0.
Low Qual Fin SF	0.026532	0.047204	0.059254	-0.042646	-0.016875	-0.050242	0.
Gr Liv Area	-0.023809	-0.165111	0.186612	0.361933	0.418321	0.577780	-0.
Bsmt Full Bath	-0.041665	-0.009074	-0.040172	0.103704	0.106726	0.148886	-0.
Bsmt Half Bath	0.019375	0.035377	0.004709	-0.029865	0.009234	-0.047082	0.
Full Bath	-0.048754	-0.235566	0.198671	0.209843	0.224679	0.556720	-0.
Half Bath	-0.036492	-0.186542	0.266447	0.077310	0.128025	0.294043	-0.
Bedroom AbvGr	0.014230	-0.014036	0.056779	0.288409	0.298550	0.077886	-0.
Kitchen AbvGr	-0.015031	0.084954	0.269444	0.010360	-0.025662	-0.170318	-0.

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

	Order	PID	MS SubClass	Lot Frontage	Lot Area	Overall Qual	(
TotRms AbvGrd	0.001629	-0.130301	0.137183	0.362169	0.383795	0.378023	-0.
Fireplaces	-0.022008	-0.151539	0.010696	0.248728	0.311907	0.419263	-0.
Garage Yr Blt	-0.056214	-0.274244	0.084326	0.126930	0.073599	0.638165	-0.
Garage Cars	-0.037003	-0.276663	0.020071	0.348630	0.344115	0.611424	-0.
Garage Area	-0.034251	-0.207054	-0.047607	0.375849	0.370980	0.547140	-0.
Wood Deck SF	-0.025811	-0.092368	0.019062	0.118401	0.177609	0.290231	-0.
Open Porch SF	0.000760	-0.176972	0.030639	0.173802	0.171777	0.440433	-0.
Enclosed Porch	0.025225	0.152769	-0.005318	-0.092787	-0.042467	-0.192093	0.
3Ssn Porch	-0.015117	0.005809	-0.033196	0.010447	0.029028	0.019398	0.
Screen Porch	0.006032	0.006353	-0.039486	0.086195	0.091527	0.026212	0.
Pool Area	0.047986	0.003222	-0.001964	0.083211	0.083071	0.033057	-0.
Misc Val	-0.038106	0.018754	-0.031959	0.037856	0.073861	-0.076443	0.
Mo Sold	0.142150	-0.051290	0.014999	0.013041	0.004774	0.029163	-0.
Yr Sold	-0.977264	0.003198	-0.021873	0.003559	-0.021720	-0.017015	0.
SalePrice	-0.035703	-0.270660	0.001973	0.397980	0.429249	0.808800	-0.

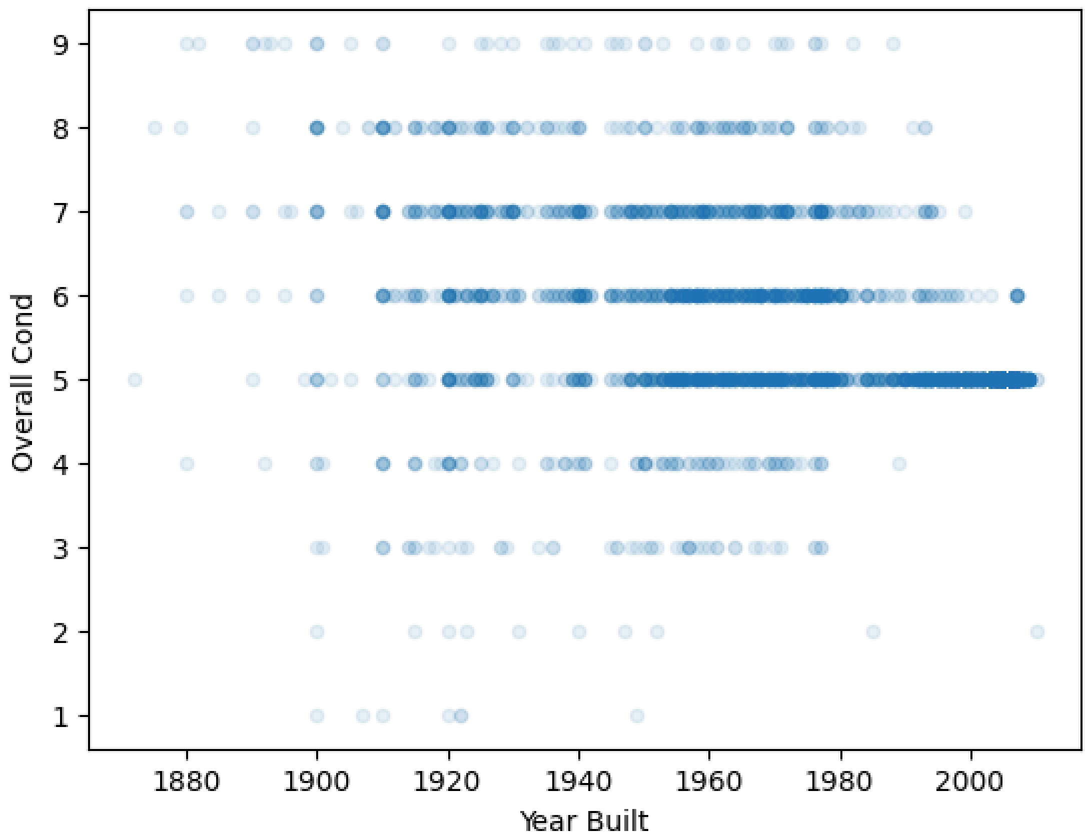
Scatter Plots

In [40]:

```
(housing
 .plot
 .scatter(x='Year Built', y='Overall Cond', alpha=.1)
)
```

Out[40]:

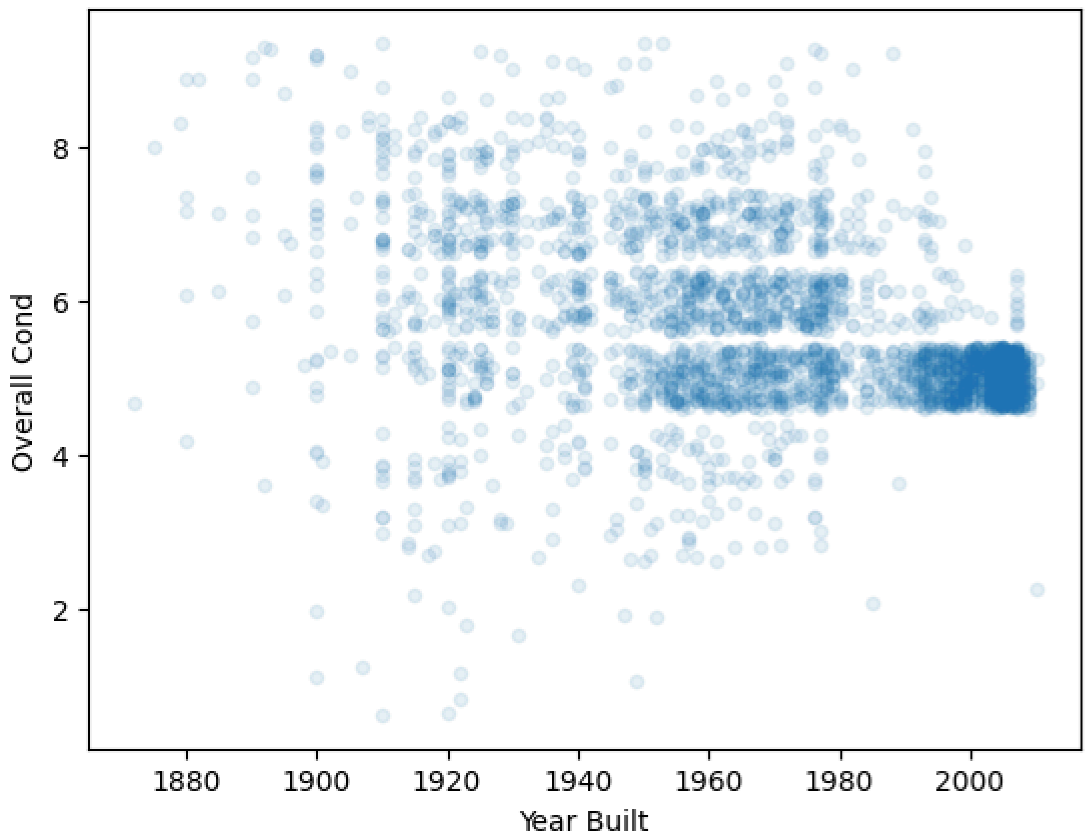
```
<Axes: xlabel='Year Built', ylabel='Overall Cond'>
```



In [41]:
```
# with jitter in y
def jitter(df_, col, amount=.5):
    return (df_
            [col] + np.random.random(len(df_))*amount - (amount/2))

(housing
 .assign(#**{'Overall Cond': housing['Overall Cond'] + np.random.random(len(
     **{'Overall Cond': jitter(housing, 'Overall Cond', amount=.8)})
 .plot
 .scatter(x='Year Built', y='Overall Cond', alpha=.1)
)
```

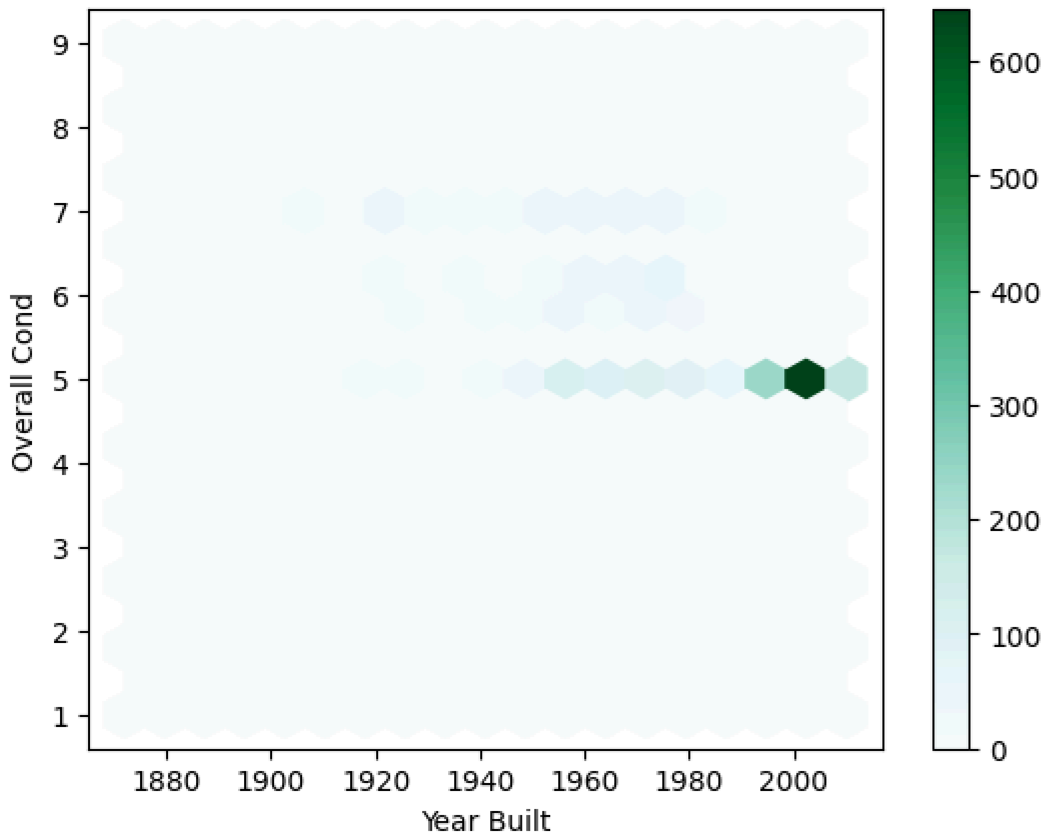
Out[41]: <Axes: xlabel='Year Built', ylabel='Overall Cond'>

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js



In [42]:
```
(housing
 .plot
 .hexbin(x='Year Built', y='Overall Cond', alpha=1, gridsize=18)
)
```

Out[42]: <Axes: xlabel='Year Built', ylabel='Overall Cond'>

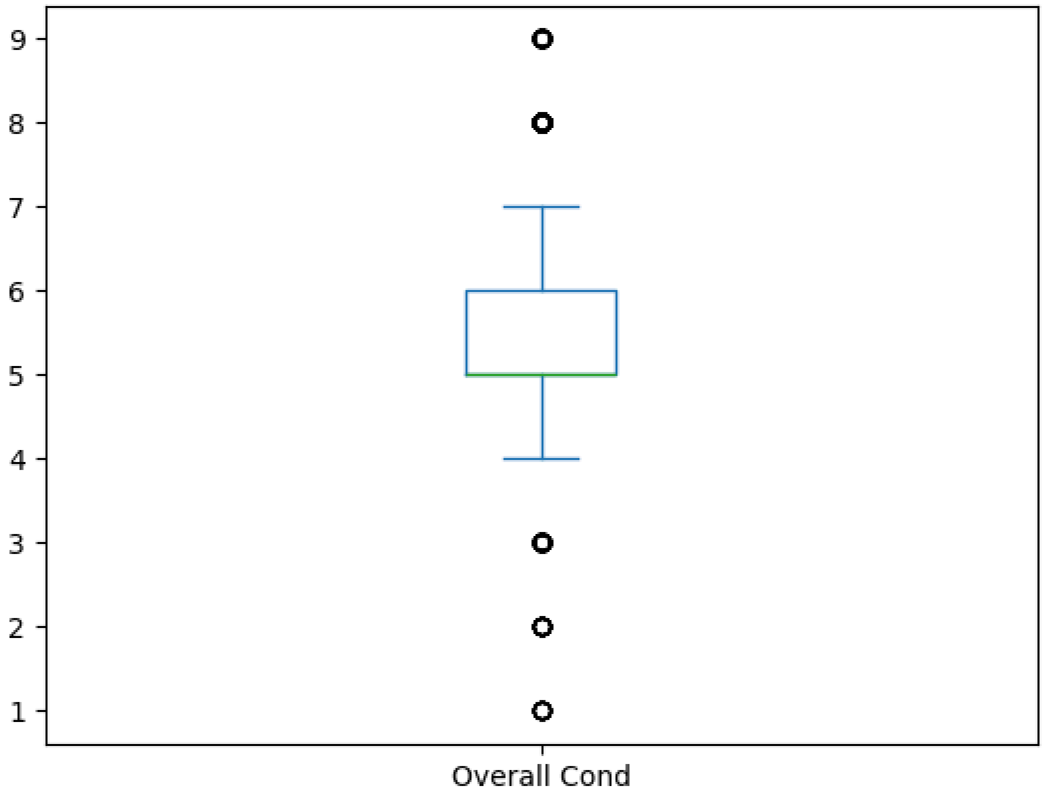


Visualizing Categoricals and Numerical Values

In [43]:
```
# Numerical and categorical
(housing
 #.assign(**{'Overall Cond': housing['Overall Cond'] + np.random.random(len(
 .plot
 .box(x='Year Built', y='Overall Cond')
)
```

Out[43]: `<Axes: >`

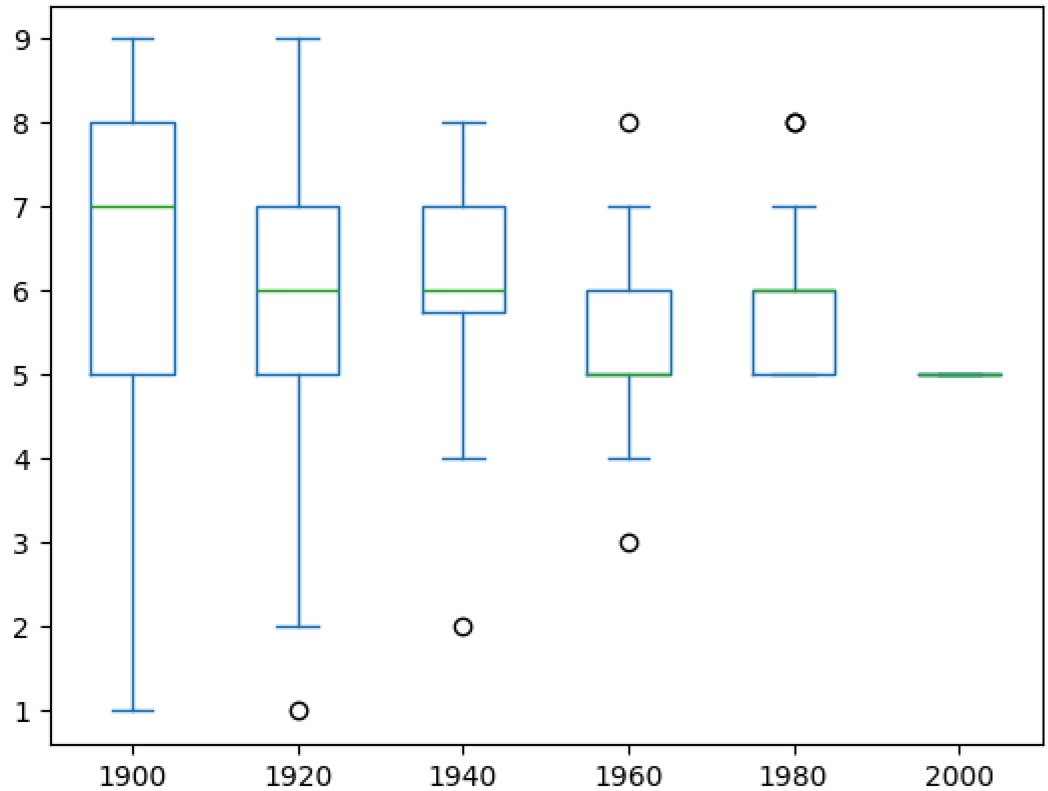
Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js



In [44]:
```
(housing
 .pivot(columns='Year Built', values='Overall Cond')
 .apply(lambda ser: ser[~ser.isna()].reset_index(drop=True))
 .loc[:, [1900, 1920, 1940, 1960, 1980, 2000]]
 .plot.box()
)
```

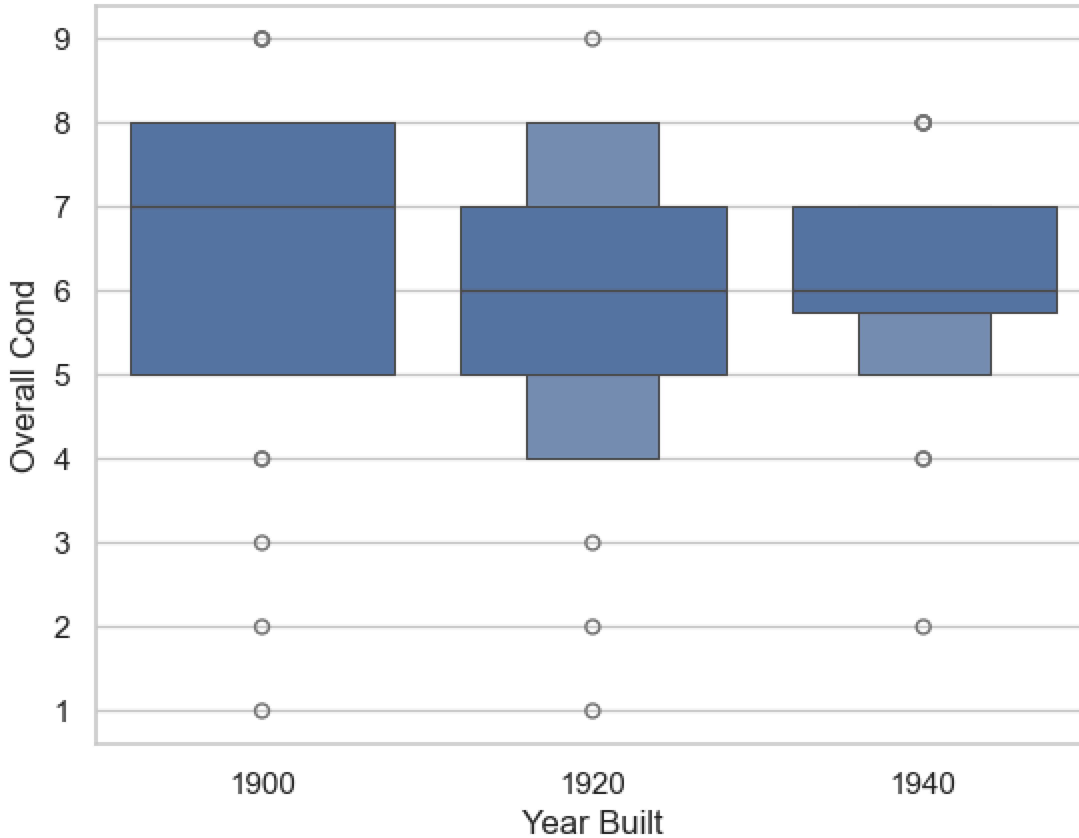
Out[44]: <Axes: >

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js



In [45]:
```
# using seaborn
sns.set(style='whitegrid')
sns.boxenplot(data=housing, x='Year Built', y='Overall Cond',
            order=[1900, 1920, 1940]
)
```

Out[45]:
```
<Axes: xlabel='Year Built', ylabel='Overall Cond'>
```



Comparing Two Categoricals

In [46]:
```
# 2 columns Categoricals - Cross tabulation
(housing
 .groupby(['Overall Qual', 'Bsmt Cond'])
 .size()
 .unstack()
)
```

```
C:\Users\deepa\AppData\Local\Temp\ipykernel_6756\1504741807.py:3: FutureWarn
ing: The default of observed=False is deprecated and will be changed to True
in a future version of pandas. Pass observed=False to retain current behavio
r or observed=True to adopt the future default and silence this warning.
  .groupby(['Overall Qual', 'Bsmt Cond'])
```

Out[46]:

Bsmt Cond	Ex	Fa	Gd	Po	TA
Overall Qual					
1	0	0	0	1	0
2	0	4	0	0	5
3	0	9	0	0	21
4	0	16	2	1	182
5	1	39	24	2	727
6	1	28	28	0	672
7	0	5	33	0	561
8	1	3	25	1	320
9	0	0	9	0	98
10	0	0	1	0	30

In [47]:

```
# using inbuilt crosstab method
(pd.crosstab(index=housing['Overall Qual'], columns=housing['Bsmt Cond'])
 .style
 .background_gradient(cmap='viridis', axis=None)  # None is whole dataframe
)
```

Out[47]:

Bsmt Cond	Ex	Fa	Gd	Po	TA
Overall Qual					
1	0	0	0	1	0
2	0	4	0	0	5
3	0	9	0	0	21
4	0	16	2	1	182
5	1	39	24	2	727
6	1	28	28	0	672
7	0	5	33	0	561
8	1	3	25	1	320
9	0	0	9	0	98
10	0	0	1	0	30

In [48]:

```

(pd.crosstab(index=housing['Overall Qual'], columns=housing['Bsmt Cond'])
 .loc[:, ['Ex', 'Gd', 'TA', 'Fa', 'Po']]
 .style
 .background_gradient(cmap='viridis', axis=None)  # None is whole dataframe
)
```

Out[48]:

Bsmt Cond Overall Qual	Ex	Gd	TA	Fa	Po
1	0	0	0	0	1
2	0	0	5	4	0
3	0	0	21	9	0
4	0	2	182	16	1
5	1	24	727	39	2
6	1	28	672	28	0
7	0	33	561	5	0
8	1	25	320	3	1
9	0	9	98	0	0
10	0	1	30	0	0

Linear Regression

In [49]:

```
def clean_housing_no_na(df):
    return (df
     .assign(**df.select_dtypes('string').replace('', 'Missing').astype('cat
             **{'Garage Yr Blt': df['Garage Yr Blt'].clip(upper=df['Year Bui
     .pipe(shrink_ints)
     .pipe(lambda df_: df_.assign(**df_.select_dtypes('number').fillna(0)))
    )
```

In [50]:

```
# need to remove na for linear regression to work
housing_with_no_na = clean_housing_no_na(raw)

X = housing_with_no_na.select_dtypes('number').drop(columns='SalePrice')
y = housing_with_no_na.SalePrice

X_train, X_test, y_train, y_test = model_selection.train_test_split(X, y, ra

lr = linear_model.LinearRegression()
lr.fit(X_train, y_train)
lr.score(X_test, y_test)
```

Out[50]: 0.8434707037243713

In [51]:

```
lr.coef_
```

Out[51]:
```
array([-1.03814738e+01,  9.02411860e-07, -1.63050576e+02,  2.81284818e+01,
        4.92668567e-01,  1.73466716e+04,  4.84079679e+03,  3.91645014e+02,
        1.76965630e+02,  2.73066661e+01,  1.05967269e+01,  4.13482965e+00,
       -2.72937988e+00,  1.20021762e+01,  1.87157913e+01,  2.54896650e+01,
       -9.18032349e+00,  3.50251329e+01,  7.35666770e+03, -1.51315712e+03,
       -1.41742224e+02, -5.39960782e+03, -7.73039749e+03, -1.43271176e+04,
        1.36191997e+03,  3.51407523e+03, -1.31044446e+01,  1.03630326e+04,
        1.49225509e+01,  1.99789208e+01, -6.04065085e+00,  2.04208107e+01,
       -5.67776073e+00,  7.08392922e+01, -3.93865793e+01, -8.71867696e+00,
        2.35118730e+02, -8.15826993e+03])
```

In [52]:
```
lr.intercept_
```

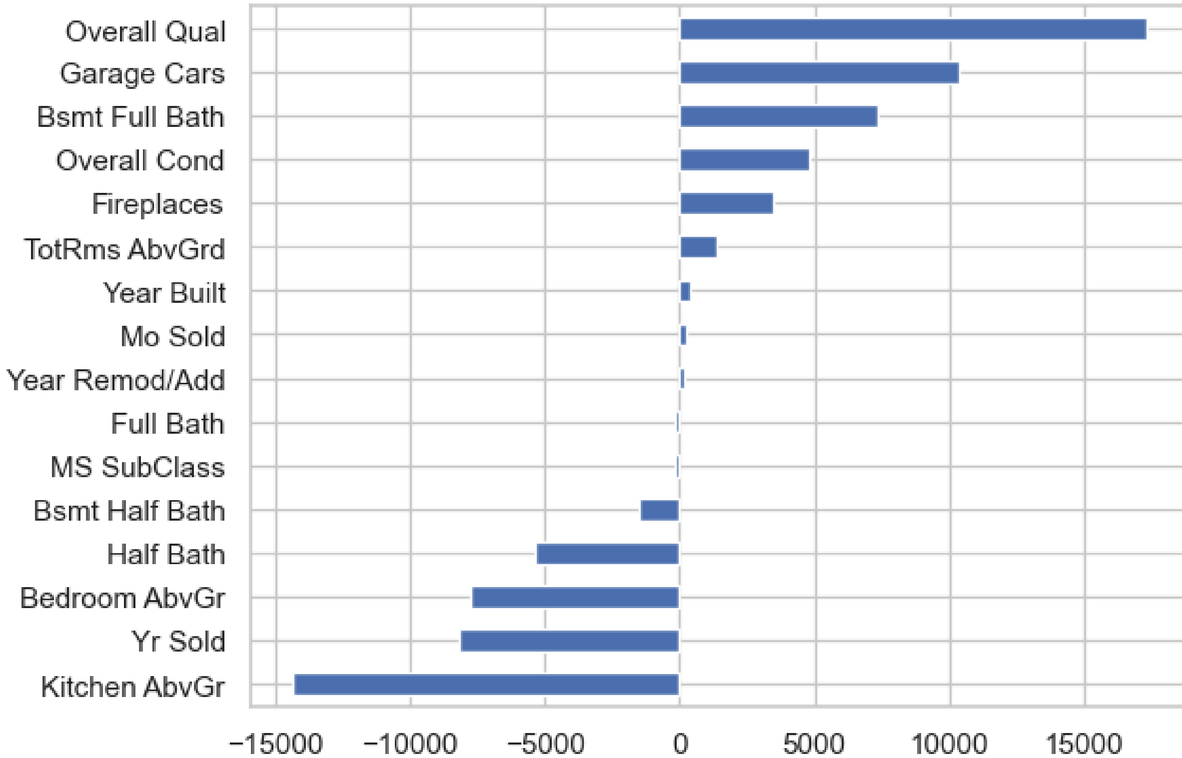
Out[52]:
```
np.float64(15240773.746057382)
```

In [53]:
```
lr.feature_names_in_
```

Out[53]:
```
array(['Order', 'PID', 'MS SubClass', 'Lot Frontage', 'Lot Area',
       'Overall Qual', 'Overall Cond', 'Year Built', 'Year Remod/Add',
       'Mas Vnr Area', 'BsmtFin SF 1', 'BsmtFin SF 2', 'Bsmt Unf SF',
       'Total Bsmt SF', '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', 'TotRms AbvGrd',
       'Fireplaces', 'Garage Yr Blt', 'Garage Cars', 'Garage Area',
       'Wood Deck SF', 'Open Porch SF', 'Enclosed Porch', '3Ssn Porch',
       'Screen Porch', 'Pool Area', 'Misc Val', 'Mo Sold', 'Yr Sold'],
      dtype=object)
```

In [54]:
```
(pd.Series(lr.coef_, index=lr.feature_names_in_)
 .pipe(lambda ser: ser[ser.abs() > 100])
 .sort_values()
 .plot.barh())
```

Out[54]:
```
<Axes: >
```



In [55]:

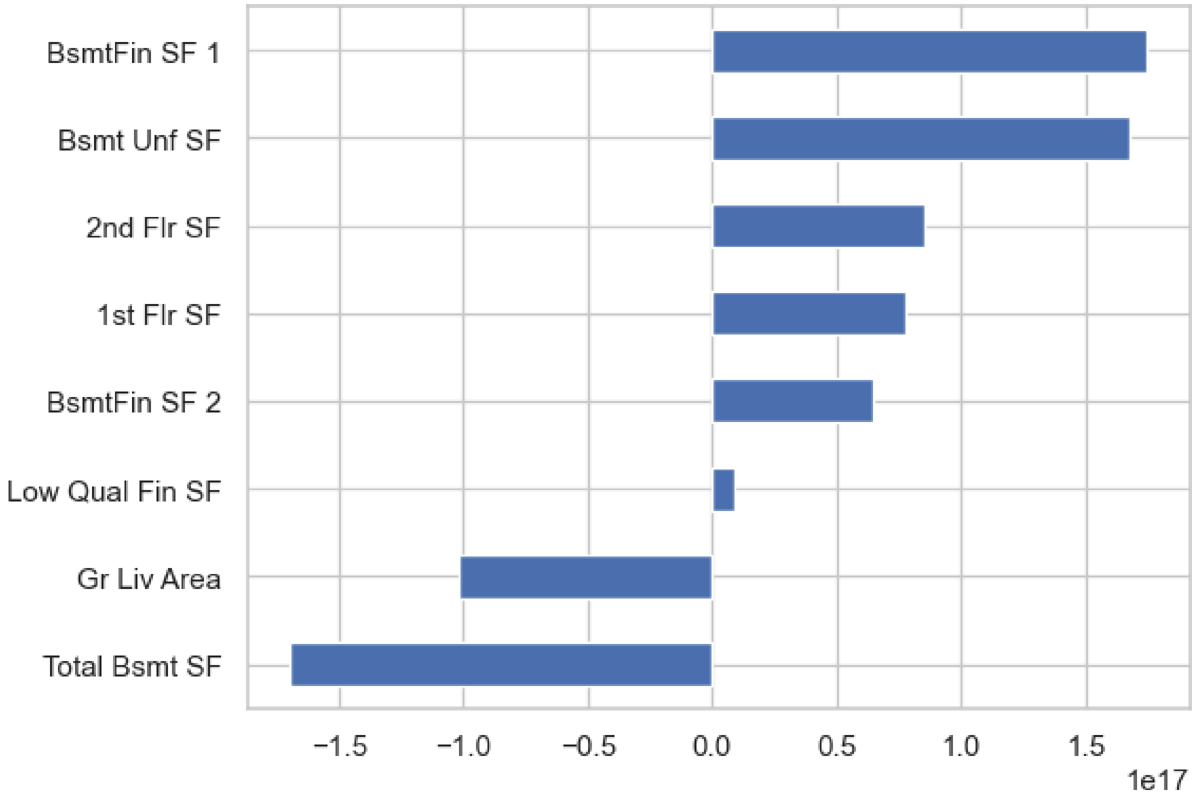
```
# with Standardizing Values

std = preprocessing.StandardScaler()
X_train = std.fit_transform(X_train)
X_test = std.transform(X_test)

lr = linear_model.LinearRegression()
lr.fit(X_train, y_train)
lr.score(X_test, y_test)
```

Out[55]: 0.843229859921004

In [56]:

```
(pd.Series(lr.coef_, index=X.columns)
 .sort_values()
 .pipe(lambda ser: ser[ser.abs() > 1e8])
 .plot.barh()
)
```

Out[56]: <Axes: >



Regression with XGBoost

In [58]:

```
# make function
def shrink_ints(df):
    mapping = {}
    for col in df.dtypes[df.dtypes=='int64[pyarrow]'].index:
        max_ = df[col].max()
        min_ = df[col].min()
        if min_ < 0:
            continue
        if max_ < 255:
            mapping[col] = 'uint8[pyarrow]'
        elif max_ < 65_535:
            mapping[col] = 'uint16[pyarrow]'
        elif max_ <  4294967295:
            mapping[col] = 'uint32[pyarrow]'
    return df.astype(mapping)


def clean_housing_no_na(df):
    return (df
     .assign(**df.select_dtypes('string').replace('', 'Missing').astype('cat
             **{'Garage Yr Blt': df['Garage Yr Blt'].clip(upper=df['Year Bui
     .pipe(shrink_ints)
     .pipe(lambda df_: df_.assign(**df_.select_dtypes('number').fillna(0)))
    )


housing2 = clean_housing_no_na(raw)
```

In [61]:
```
X = housing2.select_dtypes('number').drop(columns='SalePrice')
y = housing2.SalePrice

X_train, X_test, y_train, y_test = model_selection.train_test_split(X, y, ra

std = preprocessing.StandardScaler().set_output(transform='pandas')
X_train = std.fit_transform(X_train)
X_test = std.transform(X_test)


xg = xgb.XGBRegressor()
xg.fit(X_train, y_train)
xg.score(X_test, y_test)
```

Out[61]: 0.9202607274055481

In [62]:
```
# Use categories as well
X_cat = (housing.assign(**housing.select_dtypes('number').astype('Int64')).c
y_cat = housing.SalePrice

X_cat_train, X_cat_test, y_cat_train, y_cat_test = model_selection.train_tes

xg_cat = xgb.XGBRegressor(enable_categorical=True, tree_method='hist')

xg_cat.fit(X_cat_train, y_cat_train)
xg_cat.score(X_cat_test, y_cat_test)
```

Out[62]: 0.921321451663971

In [63]:
```
pd.Series(xg_cat.feature_importances_, index=xg_cat.feature_names_in_).sort_
```

Out[63]: <Axes: >


Overall Qual
Garage Cars
Kitchen Qual
Gr Liv Area
Central Air
Garage Cond
Fireplaces
1st Flr SF
2nd Flr SF
Neighborhood
Total Bsmt SF
Land Contour
Exter Qual
Land Slope
BsmtFin SF 1
Bsmt Full Bath
Bsmt Qual
Year Remod/Add
Sale Condition
Overall Cond
Lot Area
Screen Porch
MS Zoning
Garage Area
Fireplace Qu
Bsmt Exposure
Garage Finish
Paved Drive
Full Bath
Bsmt Cond
Exterior 2nd
Foundation
Condition 1
Open Porch SF
Garage Yr Blt
Exterior 1st
Year Built
Garage Type
Wood Deck SF
BsmtFin Type 1
Functional
Half Bath
Heating QC
Lot Config
Mas Vnr Area
Mas Vnr Type
Bsmt Unf SF
Sale Type
Bsmt Half Bath
PID
Lot Shape
Misc Feature
Bedroom AbvGr
TotRms AbvGrd
Pool QC
Low Qual Fin SF
Enclosed Porch
Alley
MS SubClass
Lot Frontage
Pool Area
Mo Sold
Exter Cond
BsmtFin SF 2
Fence
Roof Style
Misc Val
House Style
BsmtFin Type 2
Order
Electrical
3Ssn Porch
Kitchen AbvGr
Bldg Type
Condition 2
Garage Qual
Yr Sold
Heating
Roof Matl
Utilities
Street
0.4
0.6

Hypothesis Test

In [64]:
```
(housing
 .groupby('Neighborhood')
 .describe()
 .loc[['CollgCr', 'NAmes'], ['SalePrice']]
 .T
)
```

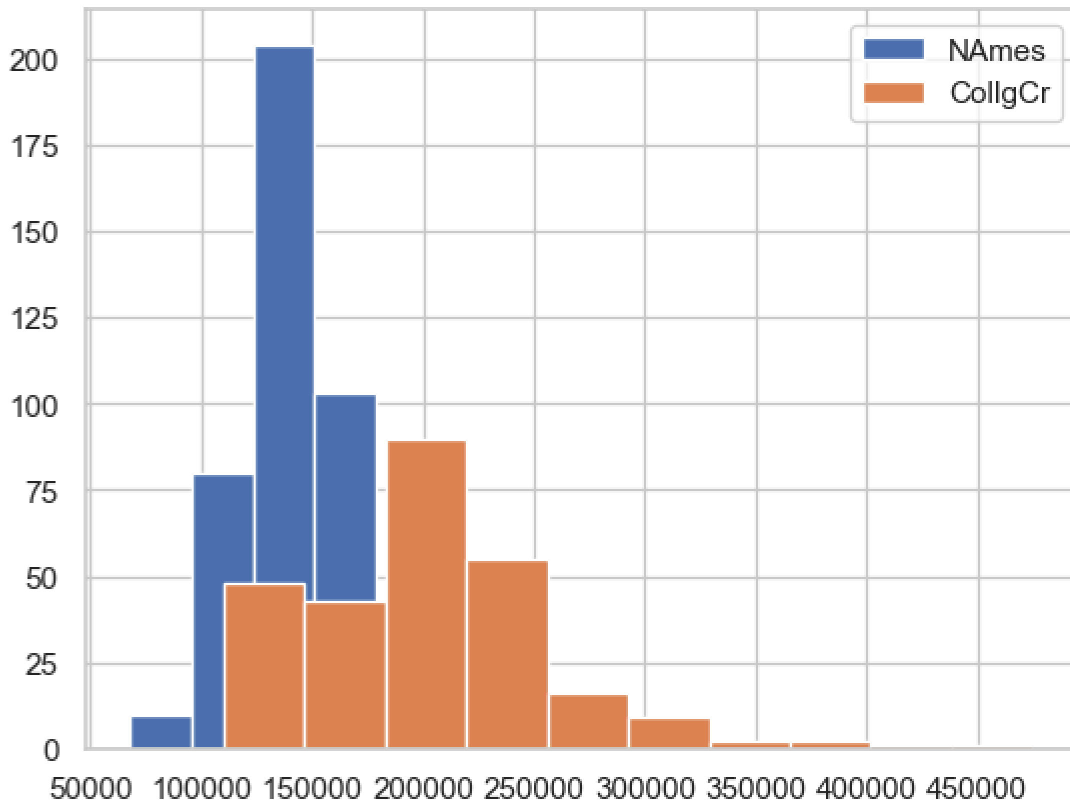
```
C:\Users\deepa\AppData\Local\Temp\ipykernel_6756\2884382520.py:2: FutureWarn
ing: The default of observed=False is deprecated and will be changed to True
in a future version of pandas. Pass observed=False to retain current behavio
r or observed=True to adopt the future default and silence this warning.
  .groupby('Neighborhood')
```

Out[64]:

	Neighborhood	CollgCr	NAmes
SalePrice	count	267.0	443.0
	mean	201803.434457	145097.349887
	std	54187.843749	31882.707229
	min	110000.0	68000.0
	25%	160875.0	127000.0
	50%	200000.0	140000.0
	75%	228250.0	157500.0
	max	475000.0	345000.0

In [65]:
```
n_ames = (housing
          .query('Neighborhood == "NAmes"')
          .SalePrice)
college_cr = (housing
          .query('Neighborhood == "CollgCr"')
          .SalePrice)

ax = n_ames.hist(label='NAmes')
college_cr.hist(ax=ax, label='CollgCr')
ax.legend()
```

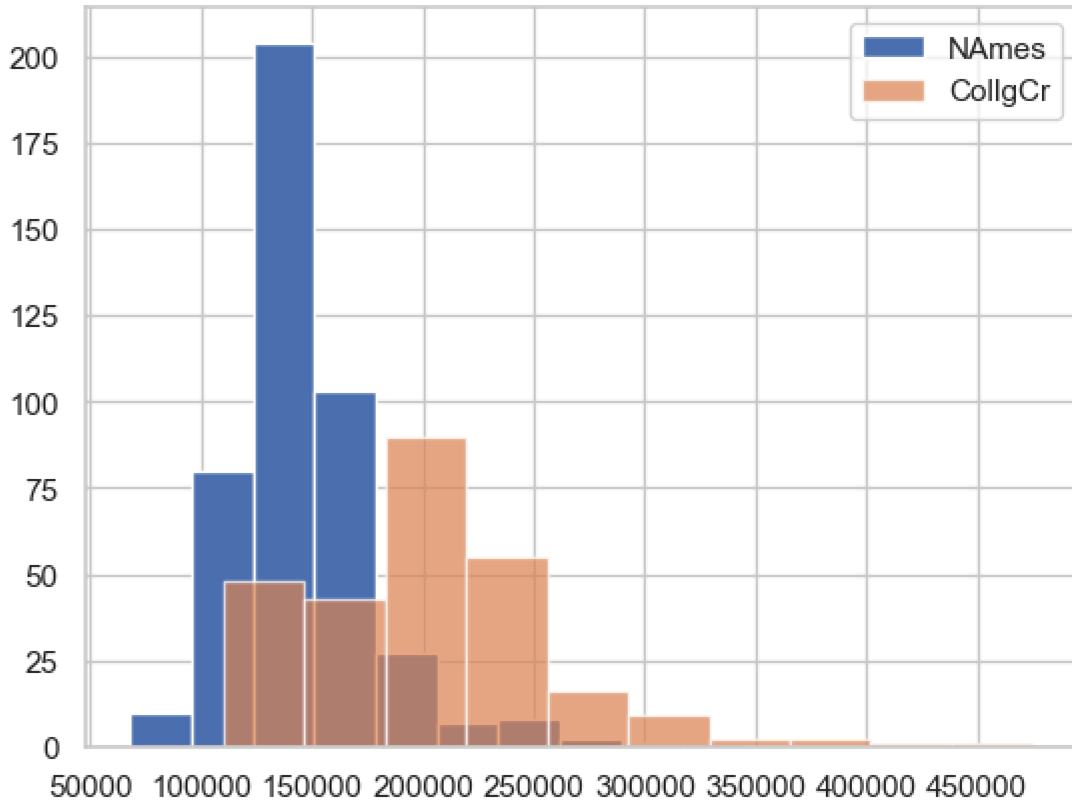
Out[65]:
```
<matplotlib.legend.Legend at 0x164c2457c80>
```



In [66]:
```
# with alpha
alpha = .7

n_ames = (housing
          .query('Neighborhood == "NAmes"')
          .SalePrice)
college_cr = (housing
          .query('Neighborhood == "CollgCr"')
          .SalePrice)

ax = n_ames.hist(label='NAmes')
college_cr.hist(ax=ax, label='CollgCr', alpha=alpha)
ax.legend()
```

Out[66]: <matplotlib.legend.Legend at 0x164c4ab16d0>

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js

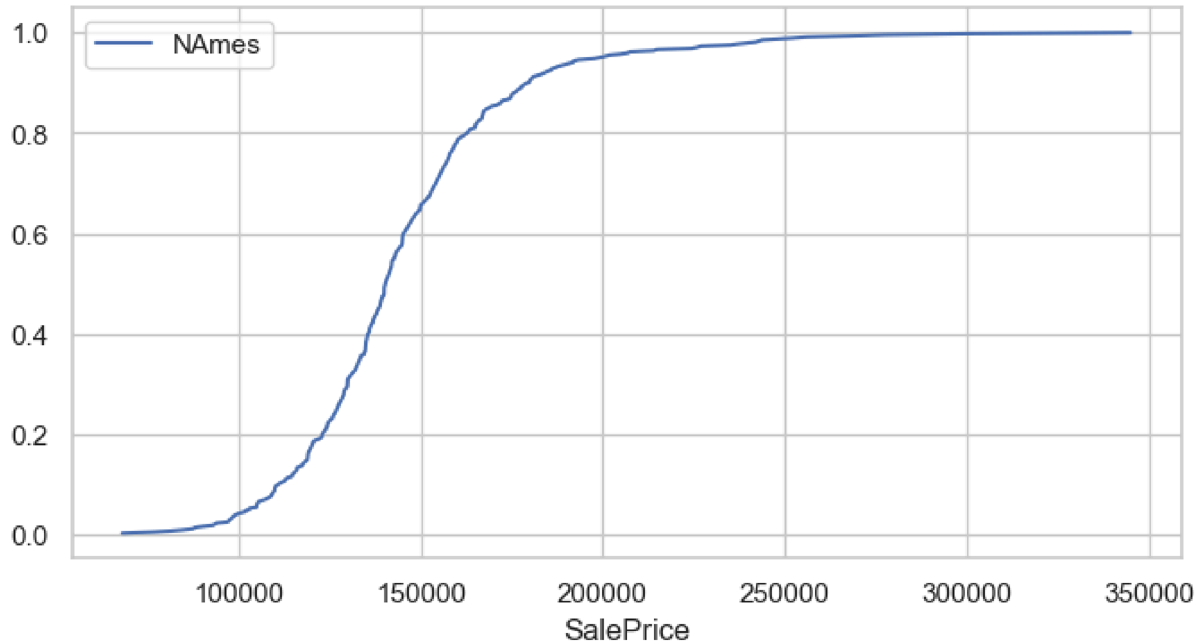


In [67]:

```
def plot_cdf(ser, ax=None, label=''):
    (ser
     .to_frame()
     .assign(cdf=ser.rank(method='average', pct=True))
     .sort_values(by='SalePrice')
     .plot(x='SalePrice', y='cdf', label=label, ax=ax)
    )
    return ser

fig, ax = plt.subplots(figsize=(8,4))
plot_cdf(n_ames, label='NAmes', ax=ax)
#plot_cdf(college_cr, label='CollegeCr', ax=ax)
```

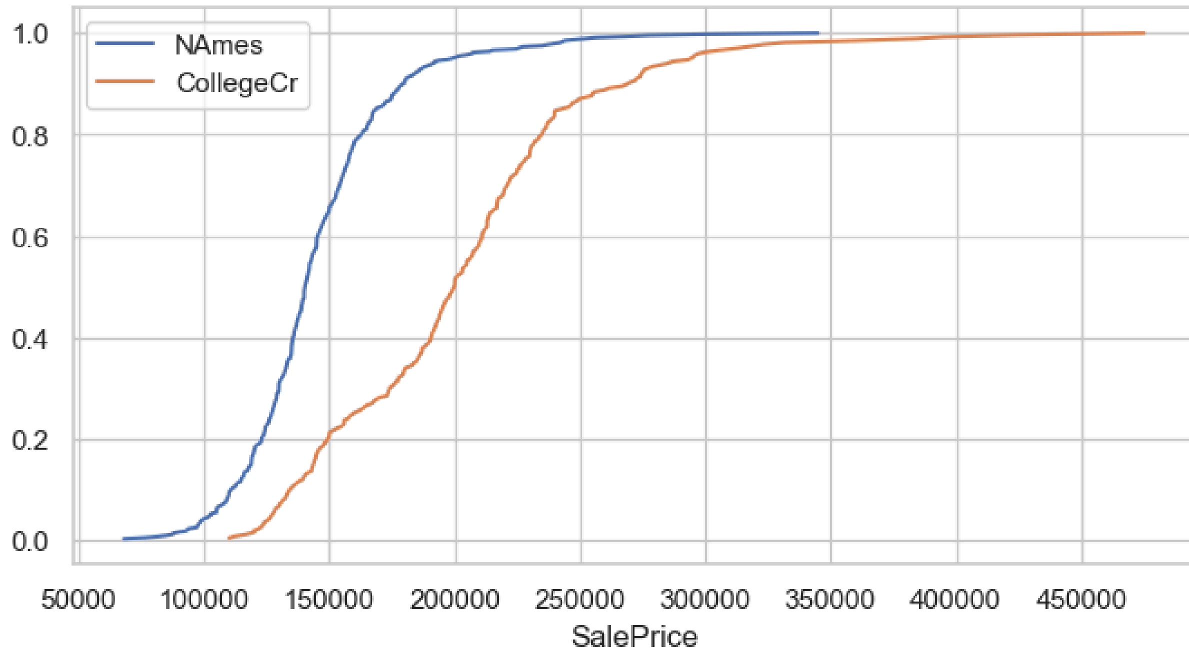
Out[67]:

```
0       215000
1       105000
2       172000
3       244000
23      149000
         ...
2630    155000
2631    134500
2632    120000
2633    105000
2634    124000
Name: SalePrice, Length: 443, dtype: uint32[pyarrow]
```



In [68]:
```
def plot_cdf(ser, ax=None, label=''):
    (ser
     .to_frame()
     .assign(cdf=ser.rank(method='average', pct=True))
     .sort_values(by='SalePrice')
     .plot(x='SalePrice', y='cdf', label=label, ax=ax)
    )
    return ser

fig, ax = plt.subplots(figsize=(8,4))
plot_cdf(n_ames, label='NAmes', ax=ax)
plot_cdf(college_cr, label='CollegeCr', ax=ax)
```

Out[68]:
```
249     245350
250     206000
251     198900
252     187000
256     159000
         ...
2811    196500
2812    198000
2813    173900
2814    163990
2815    164990
Name: SalePrice, Length: 267, dtype: uint32[pyarrow]
```



In [69]:
```
###  Running Statistical Tests
```

In [70]:
```
ks_statistic, p_value = stats.ks_2samp(n_ames, college_cr)
print(ks_statistic, p_value)
if p_value > 0.05:
    print('Fail to reject null hypothesis: Same distribution')
else:
    print('Reject null hypothesis: Not from the same distribution')
```

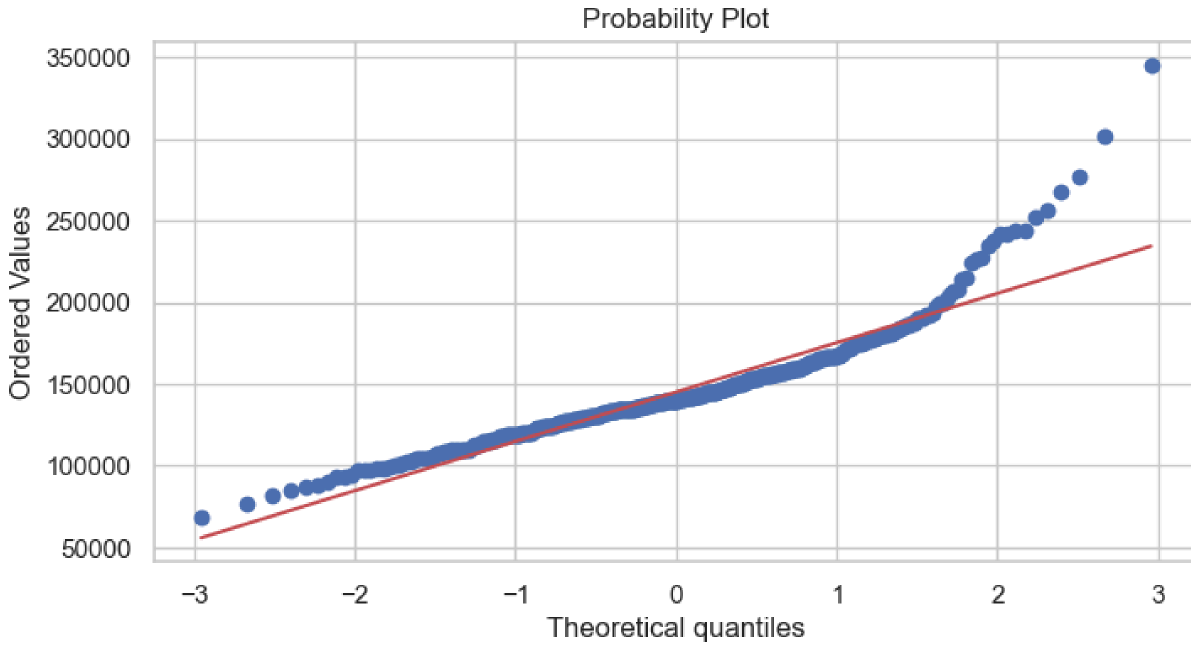
```
0.5836609430085982 3.2892428354379855e-53
Reject null hypothesis: Not from the same distribution
```

Testing for Normality

In [71]:
```
shapiro_stat, p_value = stats.shapiro(n_ames)

if p_value > 0.05:
    print("The distribution of the series is likely normal (fail to reject H
else:
    print("The distribution of the series is likely not normal (reject H0)")
```

```
The distribution of the series is likely not normal (reject H0)
```

In [72]:
```
fig, ax = plt.subplots(figsize=(8,4))
_ = stats.probplot(n_ames, plot=ax)
```


Probability Plot
Ordered Values
Theoretical quantiles
350000
300000
250000
200000
150000
100000
50000
−3
−2
−1
0
1
2
3

In [ ]: