

Engineering project

May 23, 2023

```
[1]: import numpy as np
import pandas as pd
import tensorflow as tf
import matplotlib.pyplot as plt
%matplotlib inline
from patsy import dmatrices
import sklearn
import seaborn as sns
from scipy.stats import skew, norm
```

```
[5]: dataframe=pd.read_csv("PEP1.csv")
```

```
[6]: dataframe.head()
```

```
[6]:   Id MSSubClass MSZoning LotFrontage LotArea Street Alley LotShape \
0    1          60      RL       65.0     8450    Pave  NaN    Reg
1    2          20      RL       80.0     9600    Pave  NaN    Reg
2    3          60      RL       68.0    11250    Pave  NaN    IR1
3    4          70      RL       60.0     9550    Pave  NaN    IR1
4    5          60      RL       84.0    14260    Pave  NaN    IR1

  LandContour Utilities ... PoolArea PoolQC Fence MiscFeature MiscVal MoSold \
0        Lvl    AllPub ...        0    NaN    NaN        NaN    0      2
1        Lvl    AllPub ...        0    NaN    NaN        NaN    0      5
2        Lvl    AllPub ...        0    NaN    NaN        NaN    0      9
3        Lvl    AllPub ...        0    NaN    NaN        NaN    0      2
4        Lvl    AllPub ...        0    NaN    NaN        NaN    0     12

  YrSold SaleType SaleCondition SalePrice
0  2008      WD      Normal    208500
1  2007      WD      Normal    181500
2  2008      WD      Normal    223500
3  2006      WD  Abnorml    140000
4  2008      WD      Normal    250000

[5 rows x 81 columns]
```

```
[6]: print(dataframe.shape)
```

(1460, 81)

```
[7]: null_values = dataframe.isnull()  
print(null_values)
```

```
          Id MSSubClass MSZoning LotFrontage LotArea Street Alley \
0    False    False    False    False    False  False  True
1    False    False    False    False    False  False  True
2    False    False    False    False    False  False  True
3    False    False    False    False    False  False  True
4    False    False    False    False    False  False  True
...   ...
1455  False    False    False    False    False  False  True
1456  False    False    False    False    False  False  True
1457  False    False    False    False    False  False  True
1458  False    False    False    False    False  False  True
1459  False    False    False    False    False  False  True
```

```
          LotShape LandContour Utilities ... PoolArea PoolQC Fence \
0      False    False    False ...  False   True  True
1      False    False    False ...  False   True  True
2      False    False    False ...  False   True  True
3      False    False    False ...  False   True  True
4      False    False    False ...  False   True  True
...   ...
1455  False    False    False ...  False   True  True
1456  False    False    False ...  False   True  False
1457  False    False    False ...  False   True  False
1458  False    False    False ...  False   True  True
1459  False    False    False ...  False   True  True
```

```
          MiscFeature MiscVal MoSold YrSold SaleType SaleCondition SalePrice
0        True  False  False  False  False    False  False  False
1        True  False  False  False  False    False  False  False
2        True  False  False  False  False    False  False  False
3        True  False  False  False  False    False  False  False
4        True  False  False  False  False    False  False  False
...   ...
1455  True  False  False  False  False    False  False  False
1456  True  False  False  False  False    False  False  False
1457  False  False  False  False  False    False  False  False
1458  True  False  False  False  False    False  False  False
1459  True  False  False  False  False    False  False  False
```

[1460 rows x 81 columns]

```
[8]: for column in dataframe.columns:
    unique_values = dataframe[column].unique()
    print(f"Unique values in {column}:")
    print(unique_values)
    #here I am printing the variables with unique values
```

Unique values in Id:
[1 2 3 ... 1458 1459 1460]

Unique values in MSSubClass:
[60 20 70 50 190 45 90 120 30 85 80 160 75 180 40]

Unique values in MSZoning:
['RL' 'RM' 'C (all)' 'FV' 'RH']

Unique values in LotFrontage:
[65. 80. 68. 60. 84. 85. 75. nan 51. 50. 70. 91. 72. 66.
 101. 57. 44. 110. 98. 47. 108. 112. 74. 115. 61. 48. 33. 52.
 100. 24. 89. 63. 76. 81. 95. 69. 21. 32. 78. 121. 122. 40.
 105. 73. 77. 64. 94. 34. 90. 55. 88. 82. 71. 120. 107. 92.
 134. 62. 86. 141. 97. 54. 41. 79. 174. 99. 67. 83. 43. 103.
 93. 30. 129. 140. 35. 37. 118. 87. 116. 150. 111. 49. 96. 59.
 36. 56. 102. 58. 38. 109. 130. 53. 137. 45. 106. 104. 42. 39.
 144. 114. 128. 149. 313. 168. 182. 138. 160. 152. 124. 153. 46.]

Unique values in LotArea:
[8450 9600 11250 ... 17217 13175 9717]

Unique values in Street:
['Pave' 'Grvl']

Unique values in Alley:
[nan 'Grvl' 'Pave']

Unique values in LotShape:
['Reg' 'IR1' 'IR2' 'IR3']

Unique values in LandContour:
['Lvl' 'Bnk' 'Low' 'HLS']

Unique values in Utilities:
['AllPub' 'NoSeWa']

Unique values in LotConfig:
['Inside' 'FR2' 'Corner' 'CulDSac' 'FR3']

Unique values in LandSlope:
['Gtl' 'Mod' 'Sev']

Unique values in Neighborhood:
['CollgCr' 'Veenker' 'Crawfor' 'NoRidge' 'Mitchel' 'Somerst' 'NWAmes'
 'OldTown' 'BrkSide' 'Sawyer' 'NridgHt' 'mes' 'SawyerW' 'IDOTRR' 'MeadowV'
 'Edwards' 'Timber' 'Gilbert' 'StoneBr' 'ClearCr' 'NPkVill' 'Blmngtn'
 'BrDale' 'SWISU' 'Blueste']

Unique values in Condition1:
['Norm' 'Feedr' 'PosN' 'Artery' 'RRAe' 'RRNn' 'RRAn' 'PosA' 'RRNe']

Unique values in Condition2:
['Norm' 'Artery' 'RRNn' 'Feedr' 'PosN' 'PosA' 'RRAn' 'RRAe']

Unique values in BldgType:

```

['1Fam' '2fmCon' 'Duplex' 'TwnhsE' 'Twnhs']

Unique values in HouseStyle:
['2Story' '1Story' '1.5Fin' '1.5Unf' 'SFoyer' 'SLvl' '2.5Unf' '2.5Fin']

Unique values in OverallQual:
[ 7 6 8 5 9 4 10 3 1 2]

Unique values in OverallCond:
[5 8 6 7 4 2 3 9 1]

Unique values in YearBuilt:
[2003 1976 2001 1915 2000 1993 2004 1973 1931 1939 1965 2005 1962 2006
 1960 1929 1970 1967 1958 1930 2002 1968 2007 1951 1957 1927 1920 1966
 1959 1994 1954 1953 1955 1983 1975 1997 1934 1963 1981 1964 1999 1972
 1921 1945 1982 1998 1956 1948 1910 1995 1991 2009 1950 1961 1977 1985
 1979 1885 1919 1990 1969 1935 1988 1971 1952 1936 1923 1924 1984 1926
 1940 1941 1987 1986 2008 1908 1892 1916 1932 1918 1912 1947 1925 1900
 1980 1989 1992 1949 1880 1928 1978 1922 1996 2010 1946 1913 1937 1942
 1938 1974 1893 1914 1906 1890 1898 1904 1882 1875 1911 1917 1872 1905]

Unique values in YearRemodAdd:
[2003 1976 2002 1970 2000 1995 2005 1973 1950 1965 2006 1962 2007 1960
 2001 1967 2004 2008 1997 1959 1990 1955 1983 1980 1966 1963 1987 1964
 1972 1996 1998 1989 1953 1956 1968 1981 1992 2009 1982 1961 1993 1999
 1985 1979 1977 1969 1958 1991 1971 1952 1975 2010 1984 1986 1994 1988
 1954 1957 1951 1978 1974]

Unique values in RoofStyle:
['Gable' 'Hip' 'Gambrel' 'Mansard' 'Flat' 'Shed']

Unique values in RoofMatl:
['CompShg' 'WdShngl' 'Metal' 'WdShake' 'Membran' 'Tar&Grv' 'Roll'
 'ClyTile']

Unique values in Exterior1st:
['VinylSd' 'MetalSd' 'Wd Sdng' 'HdBoard' 'BrkFace' 'WdShing' 'CemntBd'
 'Plywood' 'AsbShng' 'Stucco' 'BrkComm' 'AsphShn' 'Stone' 'ImStucc'
 'CBlock']

Unique values in Exterior2nd:
['VinylSd' 'MetalSd' 'Wd Shng' 'HdBoard' 'Plywood' 'Wd Sdng' 'CmentBd'
 'BrkFace' 'Stucco' 'AsbShng' 'Brk Cmn' 'ImStucc' 'AsphShn' 'Stone'
 'Other' 'CBlock']

Unique values in MasVnrType:
['BrkFace' 'None' 'Stone' 'BrkCmn' 'nan']

Unique values in MasVnrArea:
[1.960e+02 0.000e+00 1.620e+02 3.500e+02 1.860e+02 2.400e+02 2.860e+02
 3.060e+02 2.120e+02 1.800e+02 3.800e+02 2.810e+02 6.400e+02 2.000e+02
 2.460e+02 1.320e+02 6.500e+02 1.010e+02 4.120e+02 2.720e+02 4.560e+02
 1.031e+03 1.780e+02 5.730e+02 3.440e+02 2.870e+02 1.670e+02 1.115e+03
 4.000e+01 1.040e+02 5.760e+02 4.430e+02 4.680e+02 6.600e+01 2.200e+01
 2.840e+02 7.600e+01 2.030e+02 6.800e+01 1.830e+02 4.800e+01 2.800e+01
 3.360e+02 6.000e+02 7.680e+02 4.800e+02 2.200e+02 1.840e+02 1.129e+03
 1.160e+02 1.350e+02 2.660e+02 8.500e+01 3.090e+02 1.360e+02 2.880e+02
 7.000e+01 3.200e+02 5.000e+01 1.200e+02 4.360e+02 2.520e+02 8.400e+01
 6.640e+02 2.260e+02 3.000e+02 6.530e+02 1.120e+02 4.910e+02 2.680e+02]

```

```

7.480e+02 9.800e+01 2.750e+02 1.380e+02 2.050e+02 2.620e+02 1.280e+02
2.600e+02 1.530e+02 6.400e+01 3.120e+02 1.600e+01 9.220e+02 1.420e+02
2.900e+02 1.270e+02 5.060e+02 2.970e+02      nan 6.040e+02 2.540e+02
3.600e+01 1.020e+02 4.720e+02 4.810e+02 1.080e+02 3.020e+02 1.720e+02
3.990e+02 2.700e+02 4.600e+01 2.100e+02 1.740e+02 3.480e+02 3.150e+02
2.990e+02 3.400e+02 1.660e+02 7.200e+01 3.100e+01 3.400e+01 2.380e+02
1.600e+03 3.650e+02 5.600e+01 1.500e+02 2.780e+02 2.560e+02 2.250e+02
3.700e+02 3.880e+02 1.750e+02 2.960e+02 1.460e+02 1.130e+02 1.760e+02
6.160e+02 3.000e+01 1.060e+02 8.700e+02 3.620e+02 5.300e+02 5.000e+02
5.100e+02 2.470e+02 3.050e+02 2.550e+02 1.250e+02 1.000e+02 4.320e+02
1.260e+02 4.730e+02 7.400e+01 1.450e+02 2.320e+02 3.760e+02 4.200e+01
1.610e+02 1.100e+02 1.800e+01 2.240e+02 2.480e+02 8.000e+01 3.040e+02
2.150e+02 7.720e+02 4.350e+02 3.780e+02 5.620e+02 1.680e+02 8.900e+01
2.850e+02 3.600e+02 9.400e+01 3.330e+02 9.210e+02 7.620e+02 5.940e+02
2.190e+02 1.880e+02 4.790e+02 5.840e+02 1.820e+02 2.500e+02 2.920e+02
2.450e+02 2.070e+02 8.200e+01 9.700e+01 3.350e+02 2.080e+02 4.200e+02
1.700e+02 4.590e+02 2.800e+02 9.900e+01 1.920e+02 2.040e+02 2.330e+02
1.560e+02 4.520e+02 5.130e+02 2.610e+02 1.640e+02 2.590e+02 2.090e+02
2.630e+02 2.160e+02 3.510e+02 6.600e+02 3.810e+02 5.400e+01 5.280e+02
2.580e+02 4.640e+02 5.700e+01 1.470e+02 1.170e+03 2.930e+02 6.300e+02
4.660e+02 1.090e+02 4.100e+01 1.600e+02 2.890e+02 6.510e+02 1.690e+02
9.500e+01 4.420e+02 2.020e+02 3.380e+02 8.940e+02 3.280e+02 6.730e+02
6.030e+02 1.000e+00 3.750e+02 9.000e+01 3.800e+01 1.570e+02 1.100e+01
1.400e+02 1.300e+02 1.480e+02 8.600e+02 4.240e+02 1.047e+03 2.430e+02
8.160e+02 3.870e+02 2.230e+02 1.580e+02 1.370e+02 1.150e+02 1.890e+02
2.740e+02 1.170e+02 6.000e+01 1.220e+02 9.200e+01 4.150e+02 7.600e+02
2.700e+01 7.500e+01 3.610e+02 1.050e+02 3.420e+02 2.980e+02 5.410e+02
2.360e+02 1.440e+02 4.230e+02 4.400e+01 1.510e+02 9.750e+02 4.500e+02
2.300e+02 5.710e+02 2.400e+01 5.300e+01 2.060e+02 1.400e+01 3.240e+02
2.950e+02 3.960e+02 6.700e+01 1.540e+02 4.250e+02 4.500e+01 1.378e+03
3.370e+02 1.490e+02 1.430e+02 5.100e+01 1.710e+02 2.340e+02 6.300e+01
7.660e+02 3.200e+01 8.100e+01 1.630e+02 5.540e+02 2.180e+02 6.320e+02
1.140e+02 5.670e+02 3.590e+02 4.510e+02 6.210e+02 7.880e+02 8.600e+01
7.960e+02 3.910e+02 2.280e+02 8.800e+01 1.650e+02 4.280e+02 4.100e+02
5.640e+02 3.680e+02 3.180e+02 5.790e+02 6.500e+01 7.050e+02 4.080e+02
2.440e+02 1.230e+02 3.660e+02 7.310e+02 4.480e+02 2.940e+02 3.100e+02
2.370e+02 4.260e+02 9.600e+01 4.380e+02 1.940e+02 1.190e+02]

```

Unique values in ExterQual:

```
['Gd' 'TA' 'Ex' 'Fa']
```

Unique values in ExterCond:

```
['TA' 'Gd' 'Fa' 'Po' 'Ex']
```

Unique values in Foundation:

```
['PConc' 'CBlock' 'BrkTil' 'Wood' 'Slab' 'Stone']
```

Unique values in BsmtQual:

```
['Gd' 'TA' 'Ex' nan 'Fa']
```

Unique values in BsmtCond:

```
['TA' 'Gd' nan 'Fa' 'Po']
```

Unique values in BsmtExposure:

['No' 'Gd' 'Mn' 'Av' 'nan']

Unique values in BsmtFinType1:

['GLQ' 'ALQ' 'Unf' 'Rec' 'BLQ' 'nan' 'LwQ']

Unique values in BsmtFinSF1:

706	978	486	216	655	732	1369	859	0	851	906	998	737	733
578	646	504	840	188	234	1218	1277	1018	1153	1213	731	643	967
747	280	179	456	1351	24	763	182	104	1810	384	490	649	632
941	739	912	1013	603	1880	565	320	462	228	336	448	1201	33
588	600	713	1046	648	310	1162	520	108	569	1200	224	705	444
250	984	35	774	419	170	1470	938	570	300	120	116	512	567
445	695	405	1005	668	821	432	1300	507	679	1332	209	680	716
1400	416	429	222	57	660	1016	370	351	379	1288	360	639	495
288	1398	477	831	1904	436	352	611	1086	297	626	560	390	566
1126	1036	1088	641	617	662	312	1065	787	468	36	822	378	946
341	16	550	524	56	321	842	689	625	358	402	94	1078	329
929	697	1573	270	922	503	1334	361	672	506	714	403	751	226
620	546	392	421	905	904	430	614	450	210	292	795	1285	819
420	841	281	894	1464	700	262	1274	518	1236	425	692	987	970
28	256	1619	40	846	1124	720	828	1249	810	213	585	129	498
1270	573	1410	1082	236	388	334	874	956	773	399	162	712	609
371	540	72	623	428	350	298	1445	218	985	631	1280	241	690
266	777	812	786	1116	789	1056	50	1128	775	1309	1246	986	616
1518	664	387	471	385	365	1767	133	642	247	331	742	1606	916
185	544	553	326	778	386	426	368	459	1350	1196	630	994	168
1261	1567	299	897	607	836	515	374	1231	111	356	400	698	1247
257	380	27	141	991	650	521	1436	2260	719	377	1330	348	1219
783	969	673	1358	1260	144	584	554	1002	619	180	559	308	866
895	637	604	1302	1071	290	728	2	1441	943	231	414	349	442
328	594	816	1460	1324	1338	685	1422	1283	81	454	903	605	990
206	150	457	48	871	41	674	624	480	1154	738	493	1121	282
500	131	1696	806	1361	920	1721	187	1138	988	193	551	767	1186
892	311	827	543	1003	1059	239	945	20	1455	965	980	863	533
1084	1173	523	1148	191	1234	375	808	724	152	1180	252	832	575
919	439	381	438	549	612	1163	437	394	1416	422	762	975	1097
251	686	656	568	539	862	197	516	663	608	1636	784	249	1040
483	196	572	338	330	156	1390	513	460	659	364	564	306	505
932	750	64	633	1170	899	902	1238	528	1024	1064	285	2188	465
322	860	599	354	63	223	301	443	489	284	294	814	165	552
833	464	936	772	1440	748	982	398	562	484	417	699	696	896
556	1106	651	867	854	1646	1074	536	1172	915	595	1237	273	684
324	1165	138	1513	317	1012	1022	509	900	1085	1104	240	383	644
397	740	837	220	586	535	410	75	824	592	1039	510	423	661
248	704	412	1032	219	708	415	1004	353	702	369	622	212	645
852	1150	1258	275	176	296	538	1157	492	1198	1387	522	658	1216
1480	2096	1159	440	1456	883	547	788	485	340	1220	427	344	756
1540	666	803	1000	885	1386	319	534	125	1314	602	192	593	804
1053	532	1158	1014	194	167	776	5644	694	1572	746	1406	925	482
189	765	80	1443	259	735	734	1447	548	315	1282	408	309	203

```

865 204 790 1320 769 1070 264 759 1373 976 781 25 1110 404
580 678 958 1336 1079 49 830]

```

Unique values in BsmtFinType2:

```
['Unf' 'BLQ' 'nan' 'ALQ' 'Rec' 'LwQ' 'GLQ']
```

Unique values in BsmtFinSF2:

```

[ 0   32  668  486   93  491  506  712  362   41  169  869  150  670
 28 1080  181  768  215  374  208  441  184  279  306  180  580  690
 692 228  125 1063  620  175  820 1474  264  479  147  232  380  544
 294 258  121  391  531  344  539  713  210  311 1120  165  532  96
 495 174 1127  139  202  645  123  551  219  606  612  480  182  132
 336 468  287   35  499  723  119   40  117  239   80  472   64 1057
 127 630  128  377  764  345 1085  435  823  500  290  324  634  411
 841 1061 466  396  354  149  193  273  465  400  682  557  230  106
 791 240  547  469  177  108  600  492  211  168 1031  438  375  144
  81 906  608  276  661   68  173  972  105  420  546  334  352  872
 110 627  163 1029]

```

Unique values in BsmtUnfSF:

```

[ 150 284 434  540  490   64  317  216  952  140  134  177  175 1494
 520 832 426     0  468  525 1158  637 1777  200  204 1566  180  486
 207 649 1228 1234  380  408 1117 1097   84  326  445  383  167  465
1296  83 1632  736  192  612  816   32  935  321  860 1410  148  217
 530 1346 576  318 1143 1035  440  747  701  343  280  404  840  724
 295 1768 448   36 1530 1065  384 1288  684 1013  402  635  163  168
 176 370  350  381  410  741 1226 1053  641  516  793 1139  550  905
 104 310  252 1125  203  728  732  510  899 1362   30  958  556  413
 479 297 658  262  891 1304  519 1907  336  107  432  403  811  396
 970 506 884  400  896  253  409   93 1200  572  774  769 1335  340
 882 779 112  470  294 1686  360  441  354  700  725  320  554  312
 968 504 1107 577  660   99  871  474  289  600  755  625 1121  276
 186 1424 1140 375   92  305 1176   78  274  311  710  686  457 1232
1498 1010 160 2336  630  638  162   70 1357 1194  773  483  235  125
1390 594 1694 488  357  626  916 1020 1367  798  452  392  975  361
 270 602 1482 680  606   88  342  212 1095   96  628 1560  744 2121
 768 386 1468 1145  244  698 1079  570  476  131  184  143 1092  324
1541 1470 536  319  599  622  179  292  286   80  712  291  153 1088
1249 166 906  604  100  818  844  596  210 1603  115  103  673  726
 995 967 721 1656  972  460  208  191  438 1869  371  624  552  322
 598 268 130  484  785  733  953  847  333 1580  411  982  808 1293
 939 784 595  229  114  522  735  405  117  961 1286  672 1141  806
 165 1064 1063 245 1276  892 1008  499 1316  463  242  444  281   35
 356 988 580  651  619  544  387  901  926  135  648   75  788 1307
1078 1258 273 1436  557  930  780  813  878  122  248  588  524  288
 389 424 1375 1626  406  298 2153  417  739  225  611  237  290  264
 238 363 190 1969  697  414  316  466  420  254  960  397 1191  548
  50 178 1368 169  748  689 1264  467  605 1257  551  678  707  880
 378 223 578  969  379  765  149  912  620 1709  132  993  197 1374
   90 195 706 1163  367 1122 1515   55 1497  450  846   23  390  861
 285 1050 331 2042 1237  113  742  924  512  119  314  308  293  537

```

126	427	309	914	173	1774	823	485	1116	978	636	564	108	1184
796	366	300	542	645	664	756	247	776	849	1392	38	1406	111
545	121	2046	161	261	567	1195	874	1342	151	989	1073	927	219
224	526	1164	761	461	876	859	171	718	138	941	464	250	72
508	1584	415	82	948	893	864	1349	76	487	652	1240	801	279
1030	348	234	1198	740	89	586	323	1836	480	456	1935	338	1594
102	374	1413	491	1129	255	1496	650	1926	154	999	1734	124	1417
15	834	1649	936	778	1489	442	1434	352	458	1221	1099	416	1800
227	907	528	189	1273	563	372	702	1090	435	198	1372	174	1638
894	299	105	676	1120	431	218	110	795	1098	1043	481	666	142
447	783	1670	277	412	794	239	662	1072	717	546	430	422	188
266	1181	1753	964	1450	1905	1480	772	1032	220	187	29	495	640
193	196	720	918	1428	77	1266	1128	692	770	750	1442	1007	501
691	1550	1680	1330	1710	746	814	515	571	359	355	301	668	920
1055	1420	1752	304	1302	833	133	549	705	722	799	462	429	810
155	170	230	1459	1082	758	1290	1074	251	172	868	797	365	418
730	533	671	1012	1528	1005	1373	500	762	752	399	1042	40	26
932	278	459	568	1502	543	574	977	449	983	731	120	538	831
994	341	879	815	1212	866	1630	328	141	364	1380	81	303	940
764	1048	334	1689	690	792	585	473	246	1045	1405	201	14	841
1104	241	925	2002	74	661	708	1152	256	804	812	1085	344	425
1616	976	496	349	971	1393	1622	1352	1795	1017	1588	428	803	693
858	1284	1203	1652	39	539	1217	257	715	616	240	315	1351	1026
1571	156	61	95	482	1094	60	862	221	791	398	777	503	734
709	1252	656	1319	1422	560	1573	589	877	136]				

Unique values in TotalBsmtSF:

856	1262	920	756	1145	796	1686	1107	952	991	1040	1175	912	1494
1253	832	1004	0	1114	1029	1158	637	1777	1060	1566	900	1704	1484
520	649	1228	1234	1398	1561	1117	1097	1297	1057	1088	1350	840	938
1150	1752	1434	1656	736	955	794	816	1842	384	1425	970	860	1410
780	530	1370	576	1143	1947	1453	747	1304	2223	845	1086	462	672
1768	440	896	1237	1563	1065	1288	684	612	1013	990	1235	876	1214
824	680	1588	960	458	950	1610	741	1226	1053	641	789	793	1844
994	1264	1809	1028	729	1092	1125	1673	728	732	1080	1199	1362	1078
660	1008	924	992	1063	1267	1461	1907	928	864	1734	910	1490	1728
715	884	969	1710	825	1602	1200	572	774	1392	1232	1572	1541	882
1149	644	1617	1582	720	1064	1606	1202	1151	1052	2216	968	504	1188
1593	853	725	1431	855	1726	1360	755	1713	1121	1196	617	848	1424
1140	1100	1157	1212	689	1070	1436	686	798	1248	1498	1010	713	2392
630	1203	483	1373	1194	1462	894	1414	996	1694	735	540	626	948
1845	1020	1367	1444	1573	1302	1314	975	1604	963	1482	506	926	1422
802	740	1095	1385	1152	1240	1560	2121	1160	807	1468	1575	625	858
698	1079	768	795	1416	1003	702	1165	1470	2000	700	319	861	1896
697	972	2136	716	1347	1372	1249	1136	1502	1162	710	1719	1383	844
596	1056	3206	1358	943	1499	1922	1536	1208	1215	967	721	1684	536
958	1478	764	1848	1869	616	624	940	1142	1062	888	883	1394	1099
1268	953	744	608	847	683	870	1580	1856	982	1026	1293	939	784
1256	658	1041	1682	804	788	1144	961	1260	1310	1141	806	1281	1034

```

1276 1340 1344 988 651 1518 907 901 765 799 648 3094 1440 1258
915 1517 930 813 1533 872 1242 1364 588 709 560 1375 1277 1626
1488 808 547 1976 2153 1705 1833 1792 1216 999 1113 1073 954 264
1269 190 3200 866 1501 777 1218 1368 1084 2006 1244 3138 1379 1257
1452 528 2035 611 707 880 1051 1581 1838 1650 723 654 1204 1069
1709 998 993 1374 1389 1163 1122 1496 846 372 1164 1050 2042 1868
1437 742 770 1722 1814 1430 1058 908 600 965 1032 1299 1120 936
783 1822 1522 980 1116 978 1156 636 1554 1386 811 1520 1952 1766
981 1094 2109 525 776 1486 1629 1138 2077 1406 1021 1408 738 1477
2046 923 1291 1195 1190 874 551 1419 2444 1210 927 1112 1391 1800
360 1473 1643 1324 270 859 718 1176 1311 971 1742 941 1698 1584
1595 868 1153 893 1349 1337 1720 1479 1030 1318 1252 983 1860 836
1935 1614 761 1413 956 712 650 773 1926 731 1417 1024 849 1442
1649 1568 778 1489 2078 1454 1516 1067 1559 1127 1390 1273 918 1763
1090 1054 1039 1148 1002 1638 105 676 1184 1109 892 2217 1505 1059
951 2330 1670 1623 1017 1105 1001 546 480 1134 1104 1272 1316 1126
1181 1753 964 1466 925 1905 1500 585 1632 819 1616 1161 828 945
979 561 696 1330 817 1098 1428 673 1241 944 1225 1266 1128 485
1930 1396 916 822 750 1700 1007 1187 691 1574 1680 1346 985 1657
602 1022 1082 810 1504 1220 1132 1565 1338 1654 1620 1055 800 1306
1475 2524 1992 1193 973 854 662 1103 1154 942 1048 727 690 1096
1459 1251 1247 1074 1271 290 655 1463 1836 803 833 408 533 1012
1552 1005 1530 974 1567 1006 1042 1298 704 932 1219 1296 1198 959
1261 1598 1683 818 1600 2396 1624 831 1224 663 879 815 1630 2158
931 1660 559 1300 1702 1075 1361 1106 1476 1689 2076 792 2110 1405
1192 746 1986 841 2002 1332 935 1019 661 1309 1328 1085 6110 1246
771 976 1652 1278 1902 1274 1393 1622 1352 420 1795 544 1510 911
693 1284 1732 2033 570 1980 814 873 757 1108 2633 1571 984 1205
714 1746 1525 482 1356 862 839 1286 1485 1594 622 791 708 1223
913 656 1319 1932 539 1221 1542]

```

Unique values in Heating:

```
['GasA' 'GasW' 'Grav' 'Wall' '0thW' 'Floor']
```

Unique values in HeatingQC:

```
['Ex' 'Gd' 'TA' 'Fa' 'Po']
```

Unique values in CentralAir:

```
['Y' 'N']
```

Unique values in Electrical:

```
['SBrkr' 'FuseF' 'FuseA' 'FuseP' 'Mix' nan]
```

Unique values in 1stFlrSF:

```

[ 856 1262 920 961 1145 796 1694 1107 1022 1077 1040 1182 912 1494
1253 854 1004 1296 1114 1339 1158 1108 1795 1060 1600 900 1704 520
649 1228 1234 1700 1561 1132 1097 1297 1057 1152 1324 1328 884 938
1150 1752 1518 1656 736 955 794 816 1842 1360 1425 983 860 1426
780 581 1370 902 1143 2207 1479 747 1304 2223 845 885 1086 840
526 952 1072 1768 682 1337 1563 1065 804 1301 684 612 1013 990
1235 964 1260 905 680 1588 960 835 1225 1610 977 1535 1226 1053
1047 789 997 1844 1216 774 1282 2259 1436 729 1092 1125 1699 728
988 772 1080 1199 1586 958 660 1327 1721 1682 1214 1959 928 864

```

1734	910	1501	1728	970	875	896	969	1710	1252	1200	572	991	1392
1232	1572	1541	882	1149	808	1867	1707	1064	1362	1651	2158	1164	2234
968	769	901	1340	936	1217	1224	1593	1549	725	1431	855	1726	929
1713	1121	1279	865	848	720	1442	1696	1100	1180	1212	932	689	1236
810	1137	1248	1498	1010	811	2392	630	483	1555	1194	1490	894	1414
1014	798	1566	866	889	626	1222	1872	908	1375	1444	1306	1625	1302
1314	1005	1604	963	1382	1482	926	764	1422	802	1052	778	1113	1095
1363	1632	1560	2121	1156	1175	1468	1575	625	1085	858	698	1079	1148
1644	1003	975	1041	1336	1210	1675	2000	1122	1035	861	1944	697	972
793	2036	832	716	1153	1088	1372	1472	1249	1136	1553	1163	1898	803
1719	1383	1445	596	1056	1629	1358	943	1619	1922	1536	1621	1215	993
841	1684	536	1478	1848	1869	1453	616	1192	1167	1142	1352	495	790
672	1394	1268	1287	953	1120	752	1319	847	904	914	1580	1856	1007
1026	939	784	1269	658	1742	788	735	1144	876	1112	1288	1310	1165
806	1620	1166	1071	1050	1276	1028	756	1344	1602	1470	1196	707	907
1208	1412	765	827	734	694	2402	1440	1128	1258	933	1689	1888	956
679	813	1533	888	786	1242	624	1663	833	979	575	849	1277	1634
1502	1161	1976	1652	1493	2069	1718	1131	1850	1792	916	999	1073	1484
1766	886	3228	1133	899	1801	1218	1368	2020	1378	1244	3138	1266	1476
605	2515	1509	751	334	820	880	1159	1601	1838	1680	767	664	1377
915	768	825	1069	1717	1126	1006	1048	897	1557	1389	996	1134	1496
846	576	877	1320	703	1429	2042	1521	989	2028	838	1473	779	770
924	1826	1402	1647	1058	927	600	1186	1940	1029	1032	1299	1054	807
1828	1548	980	1012	1116	1520	1350	1089	1554	1411	800	1567	981	1094
1051	822	755	909	2113	525	851	1486	1686	1181	2097	1454	1465	1679
1437	738	1839	792	2046	923	1291	1668	1195	1190	874	551	1419	2444
1238	1067	1391	1800	1264	372	1824	859	1576	1178	1325	971	1698	1776
1616	1146	948	1349	1464	1720	1038	742	757	1506	1836	1690	1220	1117
1973	1204	1614	1430	1110	1342	966	976	1062	1127	1285	773	1966	1428
1075	1309	1044	686	1661	1008	944	1489	2084	1434	1160	941	1516	1559
1099	1701	1307	1456	918	1779	702	1512	1039	1002	1646	1547	1036	676
1184	1462	1155	1090	1187	954	892	1709	1712	872	2217	1505	1068	951
2364	1670	1063	1636	1020	1105	1015	1001	546	480	1229	1272	1316	1617
1098	1788	1466	925	1905	1500	1207	1188	1381	965	1168	561	696	1542
824	783	673	869	1241	1118	1407	750	691	1574	1504	985	1657	1664
1082	2898	1687	1654	1055	1803	1532	2524	1733	1992	1771	930	1526	1091
1523	1364	1130	1096	1338	1103	1154	799	893	829	1240	1459	1251	1247
1390	438	950	887	1021	1552	812	1530	974	986	1042	1298	1811	1265
1640	1432	959	1831	1261	1170	2129	818	1124	2411	949	1624	831	1622
842	663	879	815	1630	1074	2196	1283	1660	1318	1211	2136	1138	1702
1507	1361	1024	1141	1173	2076	1140	1034	2110	1405	760	1987	1104	713
2018	1968	1332	935	1357	661	1724	1573	1582	1659	4692	1246	753	1203
1294	1902	1274	1787	1061	708	1584	1334	693	1284	1172	2156	2053	992
1078	1980	1281	814	2633	1571	984	754	2117	998	1416	1746	1525	1221
741	1569	1223	962	1537	1932	1423	913	1578	2073	1256]			

Unique values in 2ndFlrSF:

[854	0	866	756	1053	566	983	752	1142	1218	668	1320	631	716
676	860	1519	530	808	977	1330	833	765	462	213	548	960	670	

1116	876	612	1031	881	790	755	592	939	520	639	656	1414	884
729	1523	728	351	688	941	1032	848	836	475	739	1151	448	896
524	1194	956	1070	1096	467	547	551	880	703	901	720	316	1518
704	1178	754	601	1360	929	445	564	882	920	518	817	1257	741
672	1306	504	1304	1100	730	689	591	888	1020	828	700	842	1286
864	829	1092	709	844	1106	596	807	625	649	698	840	780	568
795	648	975	702	1242	1818	1121	371	804	325	809	1200	871	1274
1347	1332	1177	1080	695	167	915	576	605	862	495	403	838	517
1427	784	711	468	1081	886	793	665	858	874	526	590	406	1157
299	936	438	1098	766	1101	1028	1017	1254	378	1160	682	110	600
678	834	384	512	930	868	224	1103	560	811	878	574	910	620
687	546	902	1000	846	1067	914	660	1538	1015	1237	611	707	527
1288	832	806	1182	1040	439	717	511	1129	1370	636	533	745	584
812	684	595	988	800	677	573	1066	778	661	1440	872	788	843
713	567	651	762	482	738	586	679	644	900	887	1872	1281	472
1312	319	978	1093	473	664	1540	1276	441	348	1060	714	744	1203
783	1097	734	767	1589	742	686	1128	1111	1174	787	1072	1088	1063
545	966	623	432	581	540	769	1051	761	779	514	455	1426	785
521	252	813	1120	1037	1169	1001	1215	928	1140	1243	571	1196	1038
561	979	701	332	368	883	1336	1141	634	912	798	985	826	831
750	456	602	855	336	408	980	998	1168	1208	797	850	898	1054
895	954	772	1230	727	454	370	628	304	582	1122	1134	885	640
580	1112	653	220	240	1362	534	539	650	918	933	712	1796	971
1175	743	523	1216	2065	272	685	776	630	984	875	913	464	1039
1259	940	892	725	924	764	925	1479	192	589	992	903	430	748
587	994	950	1323	732	1357	557	1296	390	1185	873	1611	457	796
908	550	989	932	358	1392	349	691	1349	768	208	622	857	556
1044	708	626	904	510	1104	830	981	870	694	1152]			

Unique values in LowQualFinSF:

```
[ 0 360 513 234 528 572 144 392 371 390 420 473 156 515 80 53 232 481
 120 514 397 479 205 384]
```

Unique values in GrLivArea:

1710	1262	1786	1717	2198	1362	1694	2090	1774	1077	1040	2324	912	1494
1253	854	1004	1296	1114	1339	2376	1108	1795	1060	1600	900	1704	520
1317	1228	1234	1700	1561	2452	1097	1297	1057	1152	1324	1328	884	938
1150	1752	2149	1656	1452	955	1470	1176	816	1842	1360	1425	1739	1720
2945	780	1158	1111	1370	2034	2473	2207	1479	747	2287	2223	845	1718
1086	1605	988	952	1285	1768	1230	2142	1337	1563	1065	1474	2417	1560
1224	1526	990	1235	964	2291	1588	960	835	1225	1610	1732	1535	1226
1818	1992	1047	789	1517	1844	1855	1430	2696	2259	2320	1458	1092	1125
3222	1456	1123	1080	1199	1586	754	958	840	1348	1053	2157	2054	1327
1721	1682	1214	1959	1852	1764	864	1734	1385	1501	1728	1709	875	2035
1344	969	1993	1252	1200	1096	1968	1947	2462	1232	2668	1541	882	1616
1355	1867	2161	1707	1382	1767	1651	2158	2060	1920	2234	968	1525	1802
1340	2082	3608	1217	1593	2727	1431	1726	3112	2229	1713	1121	1279	1310
848	1284	1442	1696	1100	2062	1212	1392	1236	1436	1954	1248	1498	2267
1552	2392	1302	2520	987	1555	1194	2794	894	1960	1414	1744	1487	1566
866	1440	2110	1872	1928	1375	1668	2144	1306	1625	1640	1314	1604	1792

2574	1316	764	1422	1511	2192	778	1113	1939	1363	2270	1632	1548	2121
2022	1982	1468	1575	1250	858	1396	1919	1716	2263	1644	1003	1558	1950
1743	1336	3493	2000	2243	1406	861	1944	972	1118	2036	1641	1432	2353
2646	1472	2596	2468	2730	1163	2978	803	1719	1383	2134	1192	1056	1629
1358	1638	1922	1536	1621	1215	1908	841	1684	1112	1577	1478	1626	2728
1869	1453	720	1595	1167	1142	1352	1924	1505	1574	1394	1268	1287	1664
752	1319	904	914	2466	1856	1800	1691	1301	1797	784	1953	1269	1184
2332	1367	1961	788	1034	1144	1812	1550	1288	672	1572	1620	1639	1680
2172	2078	1276	1028	2097	1400	2624	1134	1602	2630	1196	1389	907	1208
1412	1198	1365	630	1661	694	2402	1573	1258	1689	1888	1886	1376	1183
813	1533	1756	1590	1242	1663	1666	1203	1935	1135	1660	1277	1634	1502
1969	1072	1976	1652	970	1493	2643	1131	1850	1826	1216	999	1073	1484
2414	1304	1578	886	3228	1820	899	1218	1801	1322	1911	1378	1041	1368
2020	2119	2344	1796	2080	1294	1244	4676	2398	1266	928	2713	605	2515
1509	827	334	1347	1724	1159	1601	1838	2285	767	1496	2183	1635	768
825	2094	1069	1126	2046	1048	1446	1557	996	1674	2295	1647	2504	2132
943	1692	1109	1477	1320	1429	2042	2775	2028	838	860	1473	935	1582
2296	924	1402	1556	1904	1915	1986	2008	3194	1029	2153	1032	1120	1054
832	1828	2262	2614	980	1512	1790	1116	1520	1350	1750	1554	1411	3395
800	1387	796	1567	1518	1929	2704	1766	981	1094	1839	1665	1510	1469
2113	1486	2448	1181	1936	2380	1679	1437	1180	1476	1369	1136	1441	792
923	1291	1761	1102	1419	4316	2519	1539	1137	616	1148	1391	1164	2576
1824	729	1178	2554	2418	971	1742	1698	1776	1146	2031	948	1349	1464
2715	2256	2640	1529	1140	2098	1026	1471	1386	2531	1547	2365	1506	1714
1836	3279	1220	1117	1973	1204	1614	1603	1110	1342	2084	901	2087	1145
1062	2013	1895	1564	773	3140	1688	2822	1128	1428	1576	2138	1309	1044
1008	1052	936	1733	1489	1434	2126	1223	1829	1516	1067	1559	1099	1482
1165	1416	1701	1775	2358	1646	1445	1779	1481	2654	1426	1039	1372	1002
1949	910	2610	2224	1155	1090	2230	892	1712	1393	2217	1683	1068	951
2240	2364	1670	902	1063	1636	2057	2274	1015	2002	480	1229	2127	2200
1617	1686	2374	1978	1788	2236	1466	925	1905	1500	2069	1971	1962	2403
1381	965	1958	2872	1894	1308	1098	1095	918	2019	869	1241	2612	2290
1940	2030	1851	1050	944	691	1504	985	1657	1522	1271	1022	1082	1132
2898	1264	3082	1654	954	1803	2329	2524	2868	1771	930	1977	1989	1523
1364	2184	1991	1338	2337	1103	1154	2260	1571	1611	2521	893	1240	1740
1459	1251	1247	1088	438	950	2622	2021	1690	1658	1964	833	1012	698
1005	1530	1981	974	2210	986	1020	1868	2828	1006	1298	932	1811	1265
1580	1876	1671	2108	3627	1261	3086	2345	1343	1124	2514	4476	1130	1221
1699	1624	1804	1622	1863	1630	1074	2196	1283	1845	1902	1211	1846	2136
1490	1138	1933	1702	1507	2620	1190	1188	1784	1948	1141	1173	2076	1553
2058	1405	874	2167	1987	1166	1675	1889	2018	3447	1524	1357	1395	2447
1659	1970	2372	5642	1246	1983	2526	1708	1122	1274	2810	2599	2112	1787
1923	708	774	2792	1334	693	1861	872	2169	1913	2156	2634	3238	1865
1078	1980	2601	1738	1475	1374	2633	790	2117	1762	2784	1746	1584	1912
2482	1687	1513	1608	2093	1840	1848	1569	2450	2201	804	1537	1932	1725
2555	2007	913	1346	2073	2340	1256]							

Unique values in BsmtFullBath:

[1 0 2 3]

```

Unique values in BsmtHalfBath:
[0 1 2]
Unique values in FullBath:
[2 1 3 0]
Unique values in HalfBath:
[1 0 2]
Unique values in Bedroom:
[3 4 1 2 0 5 6 8]
Unique values in Kitchen:
[1 2 3 0]
Unique values in KitchenQual:
['Gd' 'TA' 'Ex' 'Fa']
Unique values in TotRmsAbvGrd:
[ 8 6 7 9 5 11 4 10 12 3 2 14]
Unique values in Functional:
['Typ' 'Min1' 'Maj1' 'Min2' 'Mod' 'Maj2' 'Sev']
Unique values in Fireplaces:
[0 1 2 3]
Unique values in FireplaceQu:
[nan 'TA' 'Gd' 'Fa' 'Ex' 'Po']
Unique values in GarageType:
['Attchd' 'Detchd' 'BuiltIn' 'CarPort' nan 'Basment' '2Types']
Unique values in GarageYrBlt:
[2003. 1976. 2001. 1998. 2000. 1993. 2004. 1973. 1931. 1939. 1965. 2005.
 1962. 2006. 1960. 1991. 1970. 1967. 1958. 1930. 2002. 1968. 2007. 2008.
 1957. 1920. 1966. 1959. 1995. 1954. 1953. nan 1983. 1977. 1997. 1985.
 1963. 1981. 1964. 1999. 1935. 1990. 1945. 1987. 1989. 1915. 1956. 1948.
 1974. 2009. 1950. 1961. 1921. 1900. 1979. 1951. 1969. 1936. 1975. 1971.
 1923. 1984. 1926. 1955. 1986. 1988. 1916. 1932. 1972. 1918. 1980. 1924.
 1996. 1940. 1949. 1994. 1910. 1978. 1982. 1992. 1925. 1941. 2010. 1927.
 1947. 1937. 1942. 1938. 1952. 1928. 1922. 1934. 1906. 1914. 1946. 1908.
 1929. 1933.]]

Unique values in GarageFinish:
['RFn' 'Unf' 'Fin' nan]
Unique values in GarageCars:
[2 3 1 0 4]
Unique values in GarageArea:
[ 548 460 608 642 836 480 636 484 468 205 384 736 352 840
  576 516 294 853 280 534 572 270 890 772 319 240 250 271
  447 556 691 672 498 246 0 440 308 504 300 670 826 386
  388 528 894 565 641 288 645 852 558 220 667 360 427 490
  379 297 283 509 405 758 461 400 462 420 432 506 684 472
  366 476 410 740 648 273 546 325 792 450 180 430 594 390
  540 264 530 435 453 750 487 624 471 318 766 660 470 720
  577 380 434 866 495 564 312 625 680 678 726 532 216 303
  789 511 616 521 451 1166 252 497 682 666 786 795 856 473
  398 500 349 454 644 299 210 431 438 675 968 721 336 810
  494 457 818 463 604 389 538 520 309 429 673 884 868 492
]
```

```

413 924 1053 439 671 338 573 732 505 575 626 898 529 685
281 539 418 588 282 375 683 843 552 870 888 746 708 513
1025 656 872 292 441 189 880 676 301 474 706 617 445 200
592 566 514 296 244 610 834 639 501 846 560 596 600 373
947 350 396 864 304 784 696 569 628 550 493 578 198 422
228 526 525 908 499 508 694 874 164 402 515 286 603 900
583 889 858 502 392 403 527 765 367 426 615 871 570 406
590 612 650 1390 275 452 842 816 621 544 486 230 261 531
393 774 749 364 627 260 256 478 442 562 512 839 330 711
1134 416 779 702 567 832 326 551 606 739 408 475 704 983
768 632 541 320 800 831 554 878 752 614 481 496 423 841
895 412 865 630 605 602 618 444 397 455 409 820 1020 598
857 595 433 776 1220 458 613 456 436 812 686 611 425 343
479 619 902 574 523 414 738 354 483 327 756 690 284 833
601 533 522 788 555 689 796 808 510 255 424 305 368 824
328 160 437 665 290 912 905 542 716 586 467 582 1248 1043
254 712 719 862 928 782 466 714 1052 225 234 324 306 830
807 358 186 693 482 813 995 757 1356 459 701 322 315 668
404 543 954 850 477 276 518 1014 753 1418 213 844 860 748
248 287 825 647 342 770 663 377 804 936 722 208 662 754
622 620 370 1069 372 923 192]

```

Unique values in GarageQual:

```
['TA' 'Fa' 'Gd' nan 'Ex' 'Po']
```

Unique values in GarageCond:

```
['TA' 'Fa' nan 'Gd' 'Po' 'Ex']
```

Unique values in PavedDrive:

```
['Y' 'N' 'P']
```

Unique values in WoodDeckSF:

```

[ 0 298 192 40 255 235 90 147 140 160 48 240 171 100 406 222 288 49
203 113 392 145 196 168 112 106 857 115 120 12 576 301 144 300 74 127
232 158 352 182 180 166 224 80 367 53 188 105 24 98 276 200 409 239
400 476 178 574 237 210 441 116 280 104 87 132 238 149 355 60 139 108
351 209 216 248 143 365 370 58 197 263 123 138 333 250 292 95 262 81
289 124 172 110 208 468 256 302 190 340 233 184 201 142 122 155 670 135
495 536 306 64 364 353 66 159 146 296 125 44 215 264 88 89 96 414
519 206 141 260 324 156 220 38 261 126 85 466 270 78 169 320 268 72
349 42 35 326 382 161 179 103 253 148 335 176 390 328 312 185 269 195
57 236 517 304 198 426 28 316 322 307 257 219 416 344 380 68 114 327
165 187 181 92 228 245 503 315 241 303 133 403 36 52 265 207 150 290
486 278 70 418 234 26 342 97 272 121 243 511 154 164 173 384 202 56
321 86 194 421 305 117 550 509 153 394 371 63 252 136 186 170 474 214
199 728 436 55 431 448 361 362 162 229 439 379 356 84 635 325 33 212
314 242 294 30 128 45 177 227 218 309 404 500 668 402 283 183 175 586
295 32 366 736]

```

Unique values in OpenPorchSF:

```

[ 61 0 42 35 84 30 57 204 4 21 33 213 112 102 154 159 110 90
56 32 50 258 54 65 38 47 64 52 138 104 82 43 146 75 72 70
49 11 36 151 29 94 101 199 99 234 162 63 68 46 45 122 184 120

```

```

20  24 130 205 108  80  66  48  25  96 111 106  40 114   8 136 132  62
228 60 238 260  27  74  16 198  26  83  34  55  22  98 172 119 208 105
140 168  28  39 148  12  51 150 117 250  10  81  44 144 175 195 128  76
 17  59 214 121  53 231 134 192 123  78 187  85 133 176 113 137 125 523
100 285  88 406 155  73 182 502 274 158 142 243 235 312 124 267 265  87
288 23 152 341 116 160 174 247 291  18 170 156 166 129 418 240  77 364
188 207  67  69 131 191  41 118 252 189 282 135  95 224 169 319  58  93
244 185 200  92 180 263 304 229 103 211 287 292 241 547  91  86 262 210
141 15 126 236]

```

Unique values in EnclosedPorch:

```

[ 0 272 228 205 176  87 172 102  37 144  64 114 202 128 156  44  77 192
140 180 183  39 184  40 552  30 126  96  60 150 120 112 252  52 224 234
244 268 137  24 108 294 177 218 242  91 160 130 169 105  34 248 236  32
 80 115 291 116 158 210  36 200  84 148 136 240  54 100 189 293 164 216
239 67  90  56 129  98 143  70 386 154 185 134 196 264 275 230 254  68
194 318  48  94 138 226 174  19 170 220 214 280 190 330 208 145 259  81
 42 123 162 286 168  20 301 198 221 212  50  99]

```

Unique values in 3SsnPorch:

```

[ 0 320 407 130 180 168 140 508 238 245 196 144 182 162  23 216  96 153
290 304]

```

Unique values in ScreenPorch:

```

[ 0 176 198 291 252  99 184 168 130 142 192 410 224 266 170 154 153 144
128 259 160 271 234 374 185 182  90 396 140 276 180 161 145 200 122  95
120  60 126 189 260 147 385 287 156 100 216 210 197 204 225 152 175 312
222 265 322 190 233  63  53 143 273 288 263  80 163 116 480 178 440 155
220 119 165  40]

```

Unique values in PoolArea:

```
[ 0 512 648 576 555 480 519 738]
```

Unique values in PoolQC:

```
[nan 'Ex' 'Fa' 'Gd']
```

Unique values in Fence:

```
[nan 'MnPrv' 'GdWo' 'GdPrv' 'MnWw']
```

Unique values in MiscFeature:

```
[nan 'Shed' 'Gar2' 'Othr' 'TenC']
```

Unique values in MiscVal:

```

[ 0   700   350   500   400   480   450 15500 1200   800   2000   600
 3500 1300    54   620   560 1400  8300 1150 2500]

```

Unique values in MoSold:

```
[ 2  5  9 12 10  8 11  4  1  7  3  6]
```

Unique values in YrSold:

```
[2008 2007 2006 2009 2010]
```

Unique values in SaleType:

```
['WD' 'New' 'COD' 'ConLD' 'ConLI' 'CWD' 'ConLw' 'Con' '0th']
```

Unique values in SaleCondition:

```
['Normal' 'Abnorml' 'Partial' 'AdjLand' 'Alloca' 'Family']
```

Unique values in SalePrice:

```
[208500 181500 223500 140000 250000 143000 307000 200000 129900 118000
129500 345000 144000 279500 157000 132000 149000 90000 159000 139000]
```

325300	139400	230000	154000	256300	134800	306000	207500	68500	40000
149350	179900	165500	277500	309000	145000	153000	109000	82000	160000
170000	130250	141000	319900	239686	249700	113000	127000	177000	114500
110000	385000	130000	180500	172500	196500	438780	124900	158000	101000
202500	219500	317000	180000	226000	80000	225000	244000	185000	144900
107400	91000	135750	136500	193500	153500	245000	126500	168500	260000
174000	164500	85000	123600	109900	98600	163500	133900	204750	214000
94750	83000	128950	205000	178000	118964	198900	169500	100000	115000
190000	136900	383970	217000	259500	176000	155000	320000	163990	136000
153900	181000	84500	128000	87000	150000	150750	220000	171000	231500
166000	204000	125000	105000	222500	122000	372402	235000	79000	109500
269500	254900	162500	412500	103200	152000	127500	325624	183500	228000
128500	215000	239000	163000	184000	243000	211000	501837	200100	120000
475000	173000	135000	153337	286000	315000	192000	148500	311872	104000
274900	171500	112000	143900	277000	98000	186000	252678	156000	161750
134450	210000	107000	311500	167240	204900	97000	386250	290000	106000
192500	148000	403000	94500	128200	216500	89500	185500	194500	318000
262500	110500	241500	137000	76500	276000	151000	73000	175500	179500
120500	266000	124500	201000	415298	228500	244600	179200	164700	88000
153575	233230	135900	131000	167000	142500	175000	158500	267000	149900
295000	305900	82500	360000	165600	119900	375000	188500	270000	187500
342643	354000	301000	126175	242000	324000	145250	214500	78000	119000
284000	207000	228950	377426	202900	87500	140200	151500	157500	437154
318061	95000	105900	177500	134000	280000	198500	147000	165000	162000
172400	134432	123000	61000	340000	394432	179000	187750	213500	76000
240000	81000	191000	426000	106500	129000	67000	241000	245500	164990
108000	258000	168000	339750	60000	222000	181134	149500	126000	142000
206300	275000	109008	195400	85400	79900	122500	212000	116000	90350
555000	162900	199900	119500	188000	256000	161000	263435	62383	188700
124000	178740	146500	187000	440000	251000	132500	208900	380000	297000
89471	326000	374000	164000	86000	133000	172785	91300	34900	430000
226700	289000	208300	164900	202665	96500	402861	265000	234000	106250
184750	315750	446261	200624	107500	39300	111250	272000	248000	213250
179665	229000	263000	112500	255500	121500	268000	325000	316600	135960
142600	224500	118500	146000	131500	181900	253293	369900	79500	185900
451950	138000	319000	114504	194201	217500	221000	359100	313000	261500
75500	137500	183200	105500	314813	305000	165150	139900	209500	93000
264561	274000	370878	143250	98300	205950	350000	145500	97500	197900
402000	423000	230500	173500	103600	257500	372500	159434	285000	227875
148800	392000	194700	755000	335000	108480	141500	89000	123500	138500
196000	312500	361919	213000	55000	302000	254000	179540	52000	102776
189000	130500	159500	341000	103000	236500	131400	93500	239900	299800
236000	265979	260400	275500	158900	179400	215200	337000	264132	216837
538000	134900	102000	395000	221500	175900	187100	161500	233000	107900
160200	146800	269790	143500	485000	582933	227680	135500	159950	144500
55993	157900	224900	271000	224000	183000	139500	232600	147400	237000
139950	174900	133500	189950	250580	248900	169000	200500	66500	303477
132250	328900	122900	154500	118858	142953	611657	125500	255000	154300

```

173733 75000 35311 238000 176500 145900 169990 193000 117500 184900
253000 239799 244400 150900 197500 172000 116500 214900 178900 37900
99500 182000 167500 85500 178400 336000 159895 255900 117000 395192
195000 197000 348000 173900 337500 121600 206000 232000 136905 119200
227000 203000 213490 194000 287000 293077 310000 119750 84000 315500
262280 278000 139600 556581 84900 176485 200141 185850 328000 167900
151400 91500 138800 155900 83500 252000 92900 176432 274725 134500
184100 133700 118400 212900 163900 259000 239500 94000 424870 174500
116900 201800 218000 235128 108959 233170 245350 625000 171900 154900
392500 745000 186700 104900 262000 219210 116050 271900 229456 80500
137900 367294 101800 138887 265900 248328 465000 186500 169900 171750
294000 165400 301500 99900 128900 183900 378500 381000 185750 68400
150500 281000 333168 206900 295493 111000 156500 72500 52500 155835
108500 283463 410000 156932 144152 216000 274300 466500 58500 237500
377500 246578 281213 137450 193879 282922 257000 223000 274970 182900
192140 143750 64500 394617 149700 149300 121000 179600 92000 287090
266500 142125 147500]

```

```
[9]: numerical_dataframe = dataframe.select_dtypes(include=['int', 'float'])
# Printing the numerical dataset
print("Numerical Dataset:")
print(numerical_dataframe)
```

Numerical Dataset:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	\
0	1	60	65.0	8450	7	5	
1	2	20	80.0	9600	6	8	
2	3	60	68.0	11250	7	5	
3	4	70	60.0	9550	7	5	
4	5	60	84.0	14260	8	5	
...	
1455	1456	60	62.0	7917	6	5	
1456	1457	20	85.0	13175	6	6	
1457	1458	70	66.0	9042	7	9	
1458	1459	20	68.0	9717	5	6	
1459	1460	20	75.0	9937	5	6	

	YearBuilt	YearRemodAdd	MasVnrArea	BsmtFinSF1	...	WoodDeckSF	\
0	2003	2003	196.0	706	...	0	
1	1976	1976	0.0	978	...	298	
2	2001	2002	162.0	486	...	0	
3	1915	1970	0.0	216	...	0	
4	2000	2000	350.0	655	...	192	
...	
1455	1999	2000	0.0	0	...	0	
1456	1978	1988	119.0	790	...	349	
1457	1941	2006	0.0	275	...	0	
1458	1950	1996	0.0	49	...	366	

1459	1965	1965	0.0	830	...	736	
	OpenPorchSF	EnclosedPorch	3SsnPorch	ScreenPorch	PoolArea	MiscVal	\
0	61	0	0	0	0	0	0
1	0	0	0	0	0	0	0
2	42	0	0	0	0	0	0
3	35	272	0	0	0	0	0
4	84	0	0	0	0	0	0
...
1455	40	0	0	0	0	0	0
1456	0	0	0	0	0	0	0
1457	60	0	0	0	0	0	2500
1458	0	112	0	0	0	0	0
1459	68	0	0	0	0	0	0
	MoSold	YrSold	SalePrice				
0	2	2008	208500				
1	5	2007	181500				
2	9	2008	223500				
3	2	2006	140000				
4	12	2008	250000				
...				
1455	8	2007	175000				
1456	2	2010	210000				
1457	5	2010	266500				
1458	4	2010	142125				
1459	6	2008	147500				

[1460 rows x 38 columns]

```
[10]: categorical_dataframe = dataframe.select_dtypes(include=['object'])
#printing the categorical dataset
print("\nCategorical Dataset:")
print(categorical_dataframe)
```

Categorical Dataset:

	MSZoning	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	\
0	RL	Pave	NaN	Reg		Lvl	AllPub	Inside	Gtl
1	RL	Pave	NaN	Reg		Lvl	AllPub	FR2	Gtl
2	RL	Pave	NaN	IR1		Lvl	AllPub	Inside	Gtl
3	RL	Pave	NaN	IR1		Lvl	AllPub	Corner	Gtl
4	RL	Pave	NaN	IR1		Lvl	AllPub	FR2	Gtl
...
1455	RL	Pave	NaN	Reg		Lvl	AllPub	Inside	Gtl
1456	RL	Pave	NaN	Reg		Lvl	AllPub	Inside	Gtl
1457	RL	Pave	NaN	Reg		Lvl	AllPub	Inside	Gtl
1458	RL	Pave	NaN	Reg		Lvl	AllPub	Inside	Gtl

```

1459      RL    Pave   NaN      Reg          Lvl     AllPub     Inside      Gtl
          Neighborhood Condition1 ... GarageType GarageFinish GarageQual \
0           CollgCr       Norm   ... Attchd        RFn        TA
1           Veenker       Feedr   ... Attchd        RFn        TA
2           CollgCr       Norm   ... Attchd        RFn        TA
3           Crawfor       Norm   ... Detchd        Unf        TA
4           NoRidge       Norm   ... Attchd        RFn        TA
...
1455         ...         ...   ...   ...   ...   ...
1456         Gilbert     Norm   ... Attchd        RFn        TA
1457         NWAmes     Norm   ... Attchd        Unf        TA
1458         Crawfor     Norm   ... Attchd        RFn        TA
1459         mes         Norm   ... Attchd        Unf        TA
1459         Edwards     Norm   ... Attchd        Fin        TA

          GarageCond PavedDrive PoolQC Fence MiscFeature SaleType SaleCondition
0             TA          Y     NaN     NaN     NaN      WD      Normal
1             TA          Y     NaN     NaN     NaN      WD      Normal
2             TA          Y     NaN     NaN     NaN      WD      Normal
3             TA          Y     NaN     NaN     NaN      WD      Abnorml
4             TA          Y     NaN     NaN     NaN      WD      Normal
...
1455         ...         ...   ...   ...   ...   ...
1456         TA          Y     NaN   MnPrv     NaN      WD      Normal
1457         TA          Y     NaN   GdPrv     Shed     WD      Normal
1458         TA          Y     NaN     NaN     NaN      WD      Normal
1459         TA          Y     NaN     NaN     NaN      WD      Normal

```

[1460 rows x 43 columns]

```
[15]: dataframe.fillna(dataframe.mean(), inplace=True) # Replace missing values with
      ↪column means
```

None

```
[16]: skewness = dataframe.skew()
print("Skewness of numerical variables:")
print(skewness)
```

Skewness of numerical variables:

Id	0.000000
MSSubClass	1.407657
LotFrontage	2.384950
LotArea	12.207688
OverallQual	0.216944
OverallCond	0.693067
YearBuilt	-0.613461
YearRemodAdd	-0.503562

```

MasVnrArea      2.676412
BsmtFinSF1     1.685503
BsmtFinSF2     4.255261
BsmtUnfSF      0.920268
TotalBsmtSF    1.524255
1stFlrSF       1.376757
2ndFlrSF       0.813030
LowQualFinSF   9.011341
GrLivArea      1.366560
BsmtFullBath   0.596067
BsmtHalfBath   4.103403
FullBath        0.036562
HalfBath        0.675897
Bedroom         0.211790
Kitchen         4.488397
TotRmsAbvGrd   0.676341
Fireplaces      0.649565
GarageYrBlt    -0.668175
GarageCars      -0.342549
GarageArea      0.179981
WoodDeckSF     1.541376
OpenPorchSF    2.364342
EnclosedPorch   3.089872
3SsnPorch      10.304342
ScreenPorch     4.122214
PoolArea        14.828374
MiscVal         24.476794
MoSold          0.212053
YrSold          0.096269
SalePrice       1.882876
dtype: float64

```

[]:

```

[22]: summary_stats = dataframe.describe()
print(summary_stats)

# Calculate skewness for each column
skewness = dataframe.apply(lambda x: skew(x.dropna()))
print("Skewness:")
print(skewness)

```

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	\
count	1460.000000	1460.000000	1460.000000	1460.000000	1460.000000	
mean	730.500000	56.897260	70.049958	10516.828082	6.099315	
std	421.610009	42.300571	22.024023	9981.264932	1.382997	
min	1.000000	20.000000	21.000000	1300.000000	1.000000	
25%	365.750000	20.000000	60.000000	7553.500000	5.000000	

50%	730.500000	50.000000	70.049958	9478.500000	6.000000		\
75%	1095.250000	70.000000	79.000000	11601.500000	7.000000		
max	1460.000000	190.000000	313.000000	215245.000000	10.000000		
count	OverallCond	YearBuilt	YearRemodAdd	MasVnrArea	BsmtFinSF1	...	\
mean	1460.000000	1460.000000	1460.000000	1460.000000	1460.000000	...	
std	5.575342	1971.267808	1984.865753	103.685262	443.639726	...	
min	1.112799	30.202904	20.645407	180.569112	456.098091	...	
25%	1.000000	1872.000000	1950.000000	0.000000	0.000000	...	
50%	5.000000	1954.000000	1967.000000	0.000000	0.000000	...	
75%	5.000000	1973.000000	1994.000000	0.000000	383.500000	...	
max	9.000000	2000.000000	2004.000000	164.250000	712.250000	...	
count	WoodDeckSF	OpenPorchSF	EnclosedPorch	3SsnPorch	ScreenPorch	...	\
mean	1460.000000	1460.000000	1460.000000	1460.000000	1460.000000	...	
std	94.244521	46.660274	21.954110	3.409589	15.060959	...	
min	125.338794	66.256028	61.119149	29.317331	55.757415	...	
25%	0.000000	0.000000	0.000000	0.000000	0.000000	...	
50%	0.000000	25.000000	0.000000	0.000000	0.000000	...	
75%	168.000000	68.000000	0.000000	0.000000	0.000000	...	
max	857.000000	547.000000	552.000000	508.000000	480.000000	...	
count	PoolArea	MiscVal	MoSold	YrSold	SalePrice		
mean	1460.000000	1460.000000	1460.000000	1460.000000	1460.000000		
std	2.758904	43.489041	6.321918	2007.815753	180921.195890		
min	40.177307	496.123024	2.703626	1.328095	79442.502883		
25%	0.000000	0.000000	1.000000	2006.000000	34900.000000		
50%	0.000000	0.000000	5.000000	2007.000000	129975.000000		
75%	0.000000	0.000000	6.000000	2008.000000	163000.000000		
max	738.000000	15500.000000	8.000000	2009.000000	214000.000000		
			12.000000	2010.000000	755000.000000		

[8 rows x 38 columns]

```

  ↗
  ↘-----



TypeError                                Traceback (most recent call ↗
last)
<ipython-input-22-a55470909eb6> in <module>
      3
      4 # Calculate skewness for each column
----> 5 skewness = dataframe.apply(lambda x: skew(x.dropna()))
      6 print("Skewness:")


```

```

7 print(skewness)

    /usr/local/lib/python3.7/site-packages/pandas/core/frame.py in
→apply(self, func, axis, raw, result_type, args, **kwds)
    7550         kwds=kwds,
    7551     )
→ 7552     return op.get_result()
7553
7554 def applymap(self, func) -> "DataFrame":


    /usr/local/lib/python3.7/site-packages/pandas/core/apply.py in
→get_result(self)
    183         return self.apply_raw()
    184
--> 185         return self.apply_standard()
    186
    187     def apply_empty_result(self):


    /usr/local/lib/python3.7/site-packages/pandas/core/apply.py in
→apply_standard(self)
    274
    275     def apply_standard(self):
--> 276         results, res_index = self.apply_series_generator()
    277
    278         # wrap results


    /usr/local/lib/python3.7/site-packages/pandas/core/apply.py in
→apply_series_generator(self)
    303             for i, v in enumerate(series_gen):
    304                 # ignore SettingWithCopy here in case the user
→mutates
--> 305                 results[i] = self.f(v)
    306                 if isinstance(results[i], ABCSeries):
    307                     # If we have a view on v, we need to make a
→copy because


<ipython-input-22-a55470909eb6> in <lambda>(x)
    3
    4 # Calculate skewness for each column
----> 5 skewness = datafram.apply(lambda x: skew(x.dropna()))
    6 print("Skewness:")
    7 print(skewness)

```

```

    /usr/local/lib/python3.7/site-packages/scipy/stats/stats.py in skew(a,_
→axis, bias, nan_policy)
    1156         return mstats_basic.skew(a, axis, bias)
    1157
-> 1158     m2 = moment(a, 2, axis)
    1159     m3 = moment(a, 3, axis)
    1160     zero = (m2 == 0)

    /usr/local/lib/python3.7/site-packages/scipy/stats/stats.py in moment(a,_
→moment, axis, nan_policy)
    968         return np.array(mmnt)
    969     else:
--> 970         return _moment(a, moment, axis)
    971
    972

    /usr/local/lib/python3.7/site-packages/scipy/stats/stats.py in_
→_moment(a, moment, axis)
    1008
    1009         # Starting point for exponentiation by squares
-> 1010         a_zero_mean = a - np.expand_dims(np.mean(a, axis), axis)
    1011         if n_list[-1] == 1:
    1012             s = a_zero_mean.copy()

<__array_function__ internals> in mean(*args, **kwargs)

    /usr/local/lib/python3.7/site-packages/numpy/core/fromnumeric.py in_
→mean(a, axis, dtype, out, keepdims, where)
    3439
    3440     return _methods._mean(a, axis=axis, dtype=dtype,
-> 3441                         out=out, **kwargs)
    3442
    3443

    /usr/local/lib/python3.7/site-packages/numpy/core/_methods.py in_
→_mean(a, axis, dtype, out, keepdims, where)
    189         ret = ret.dtype.type(ret / rcount)
    190     else:
--> 191         ret = ret / rcount
    192

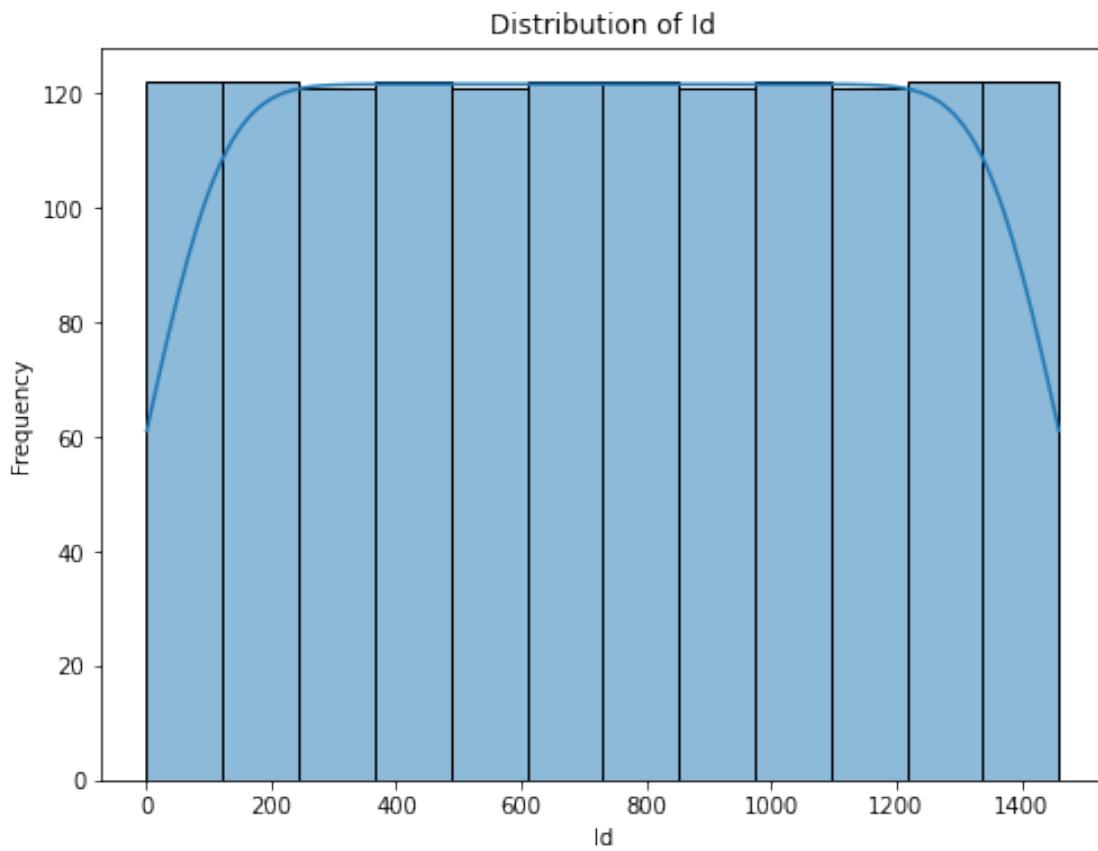
```

```
193     return ret
```

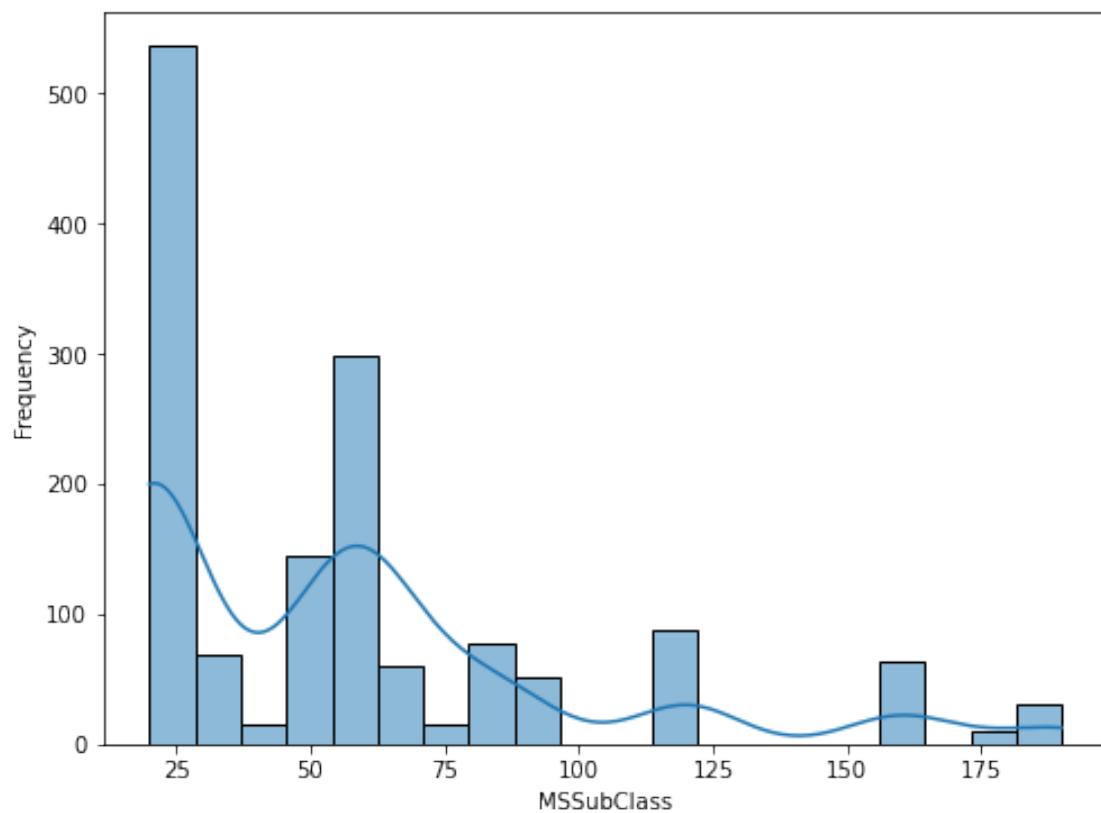
```
TypeError: ufunc 'true_divide' not supported for the input types, and  
the inputs could not be safely coerced to any supported types according to the  
casting rule 'safe'
```

```
[22]: numeric_cols = dataframe.select_dtypes(include=np.number).columns
```

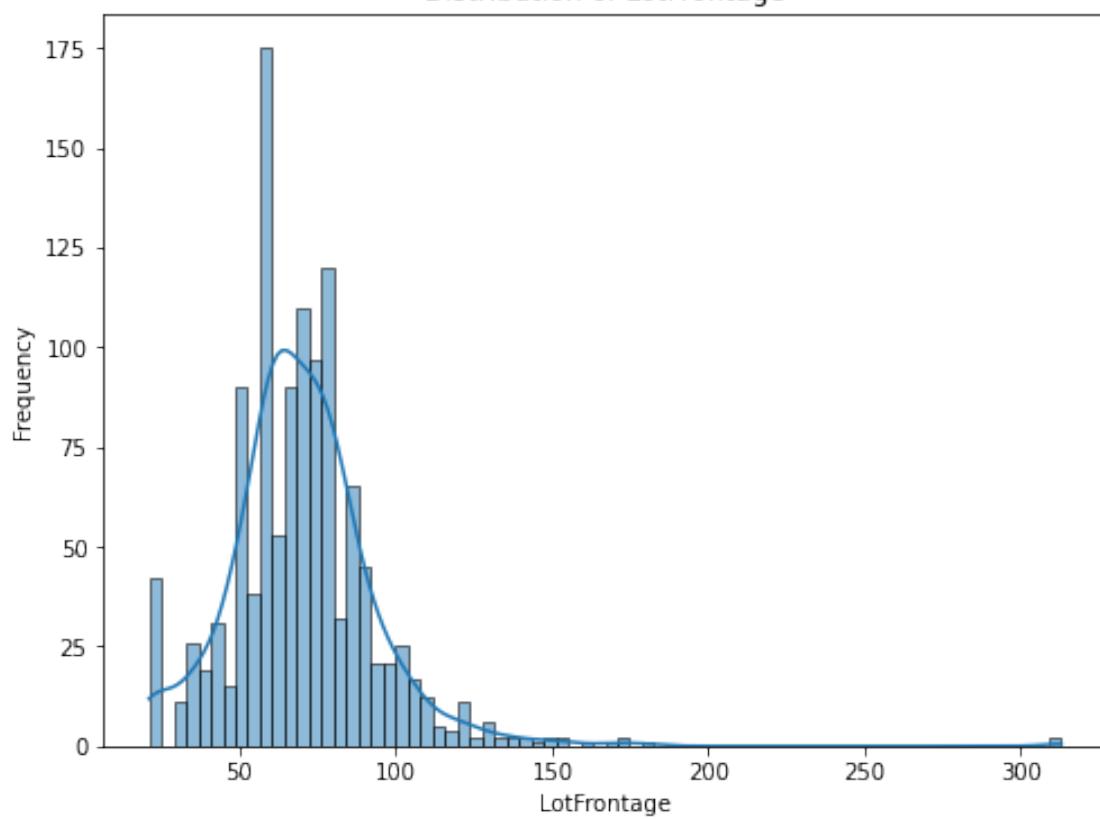
```
for col in numeric_cols:  
    plt.figure(figsize=(8, 6))  
    sns.histplot(dataframe[col].dropna(), kde=True)  
    plt.title(f'Distribution of {col}')  
    plt.xlabel(col)  
    plt.ylabel('Frequency')  
    plt.show()
```

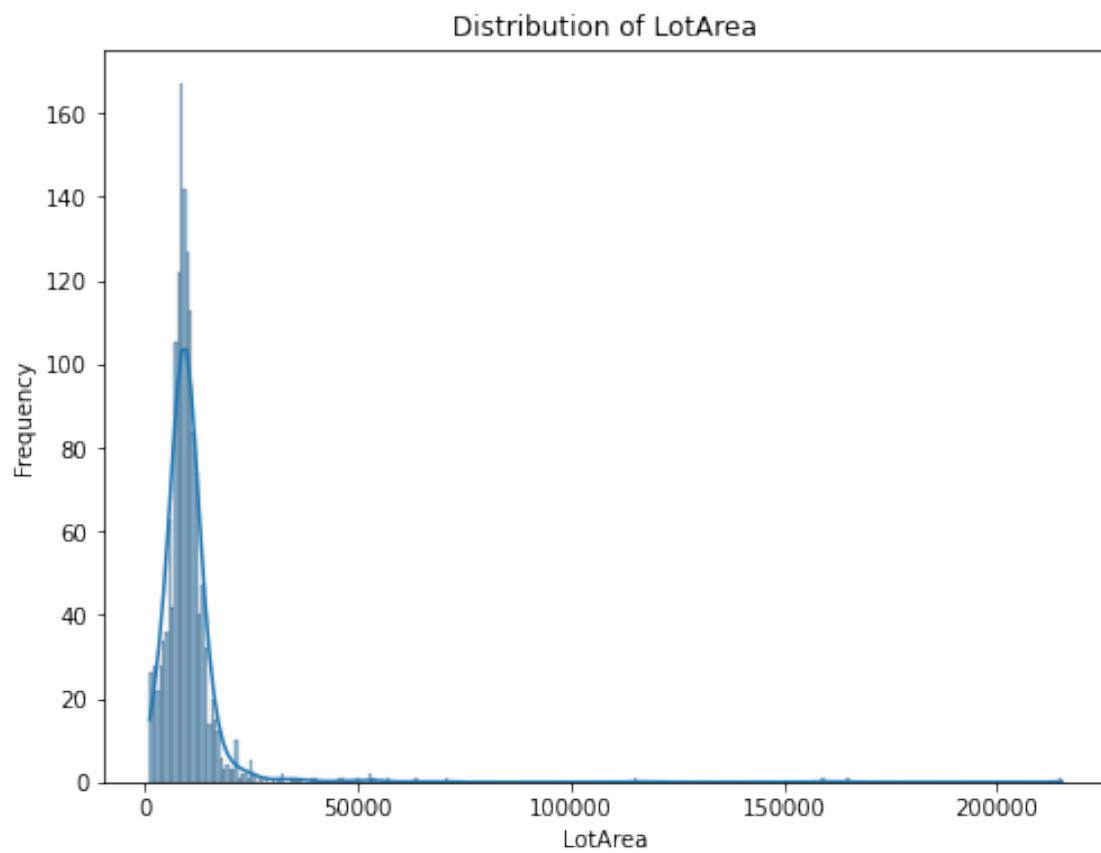


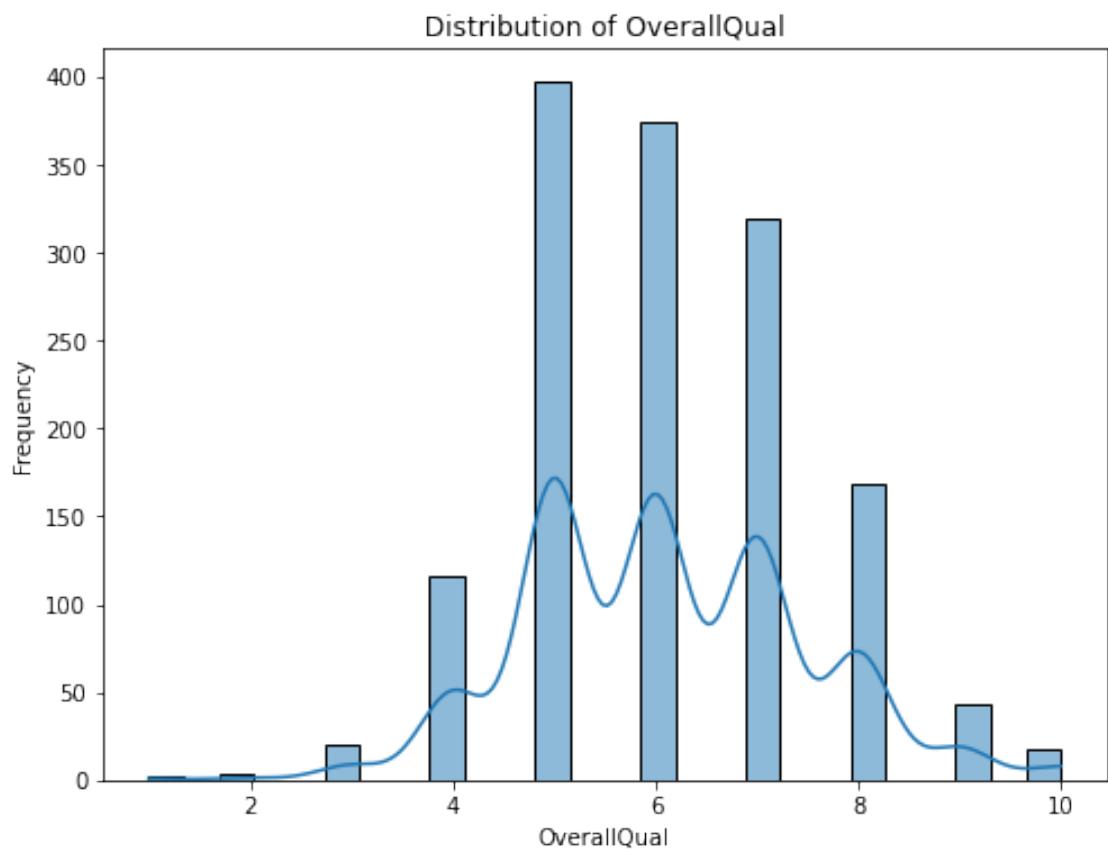
Distribution of MSSubClass

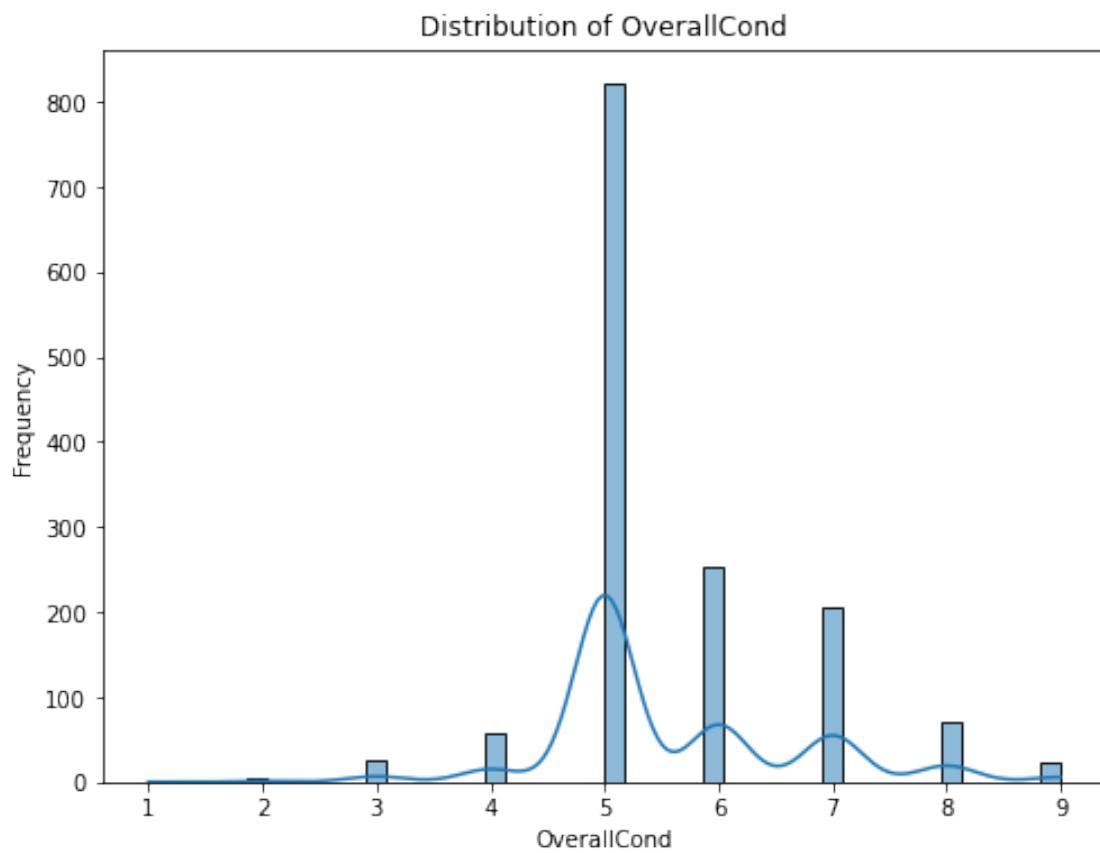


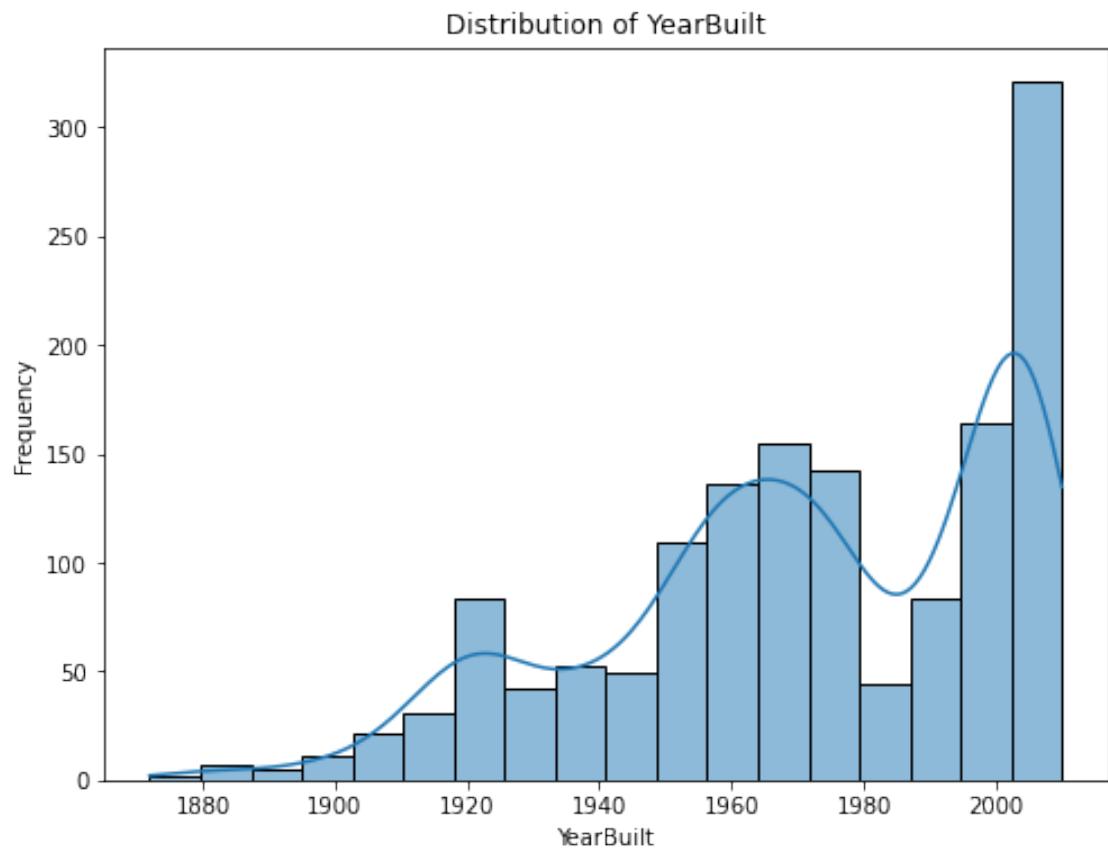
Distribution of LotFrontage

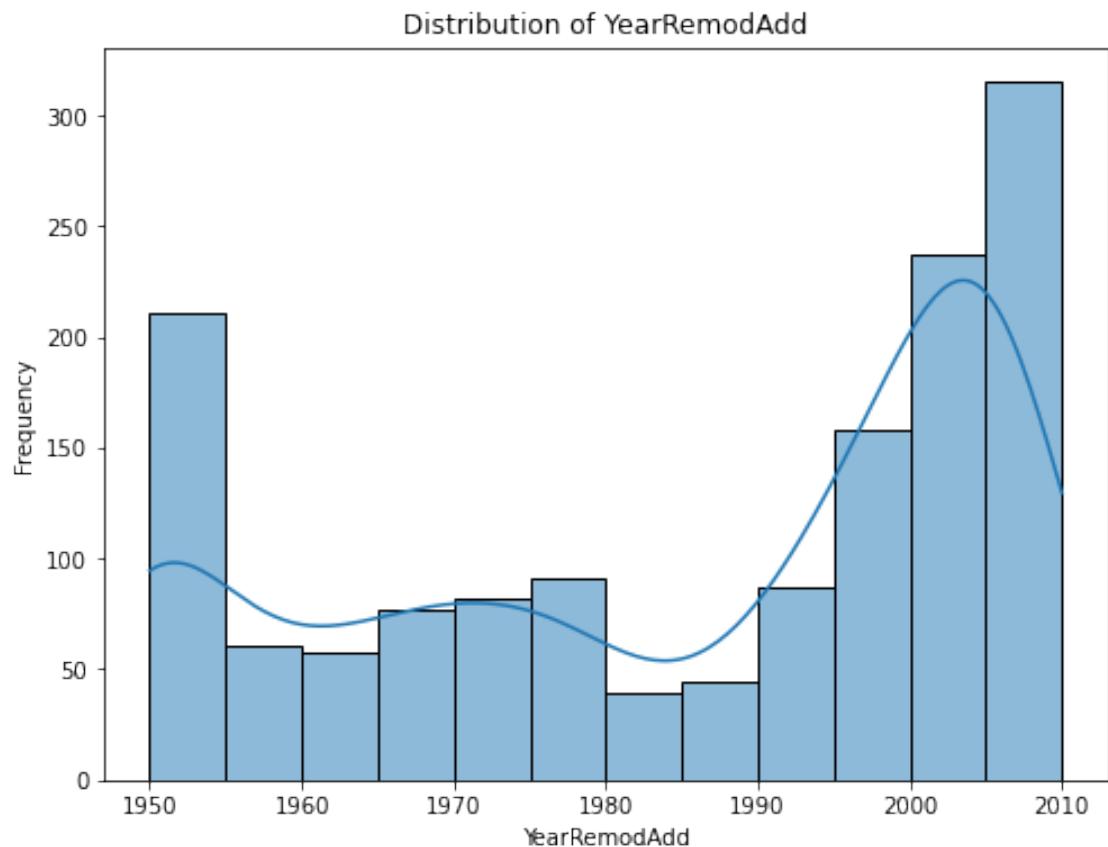




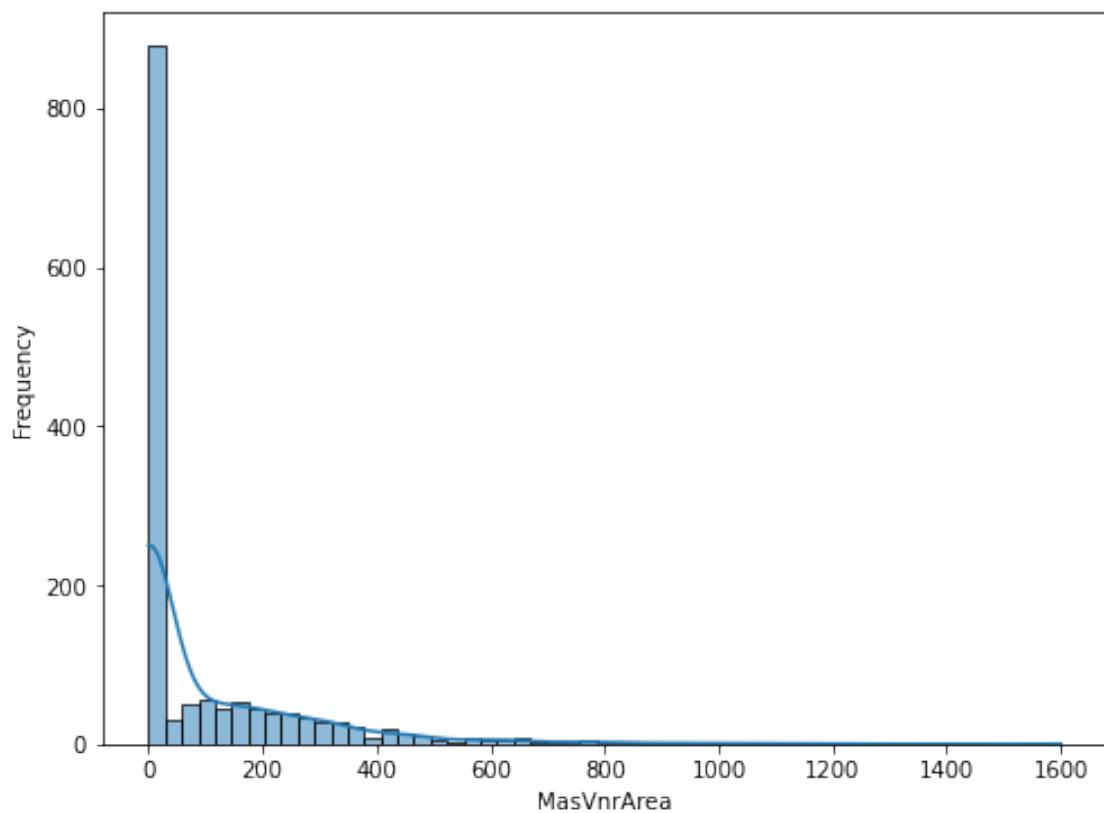




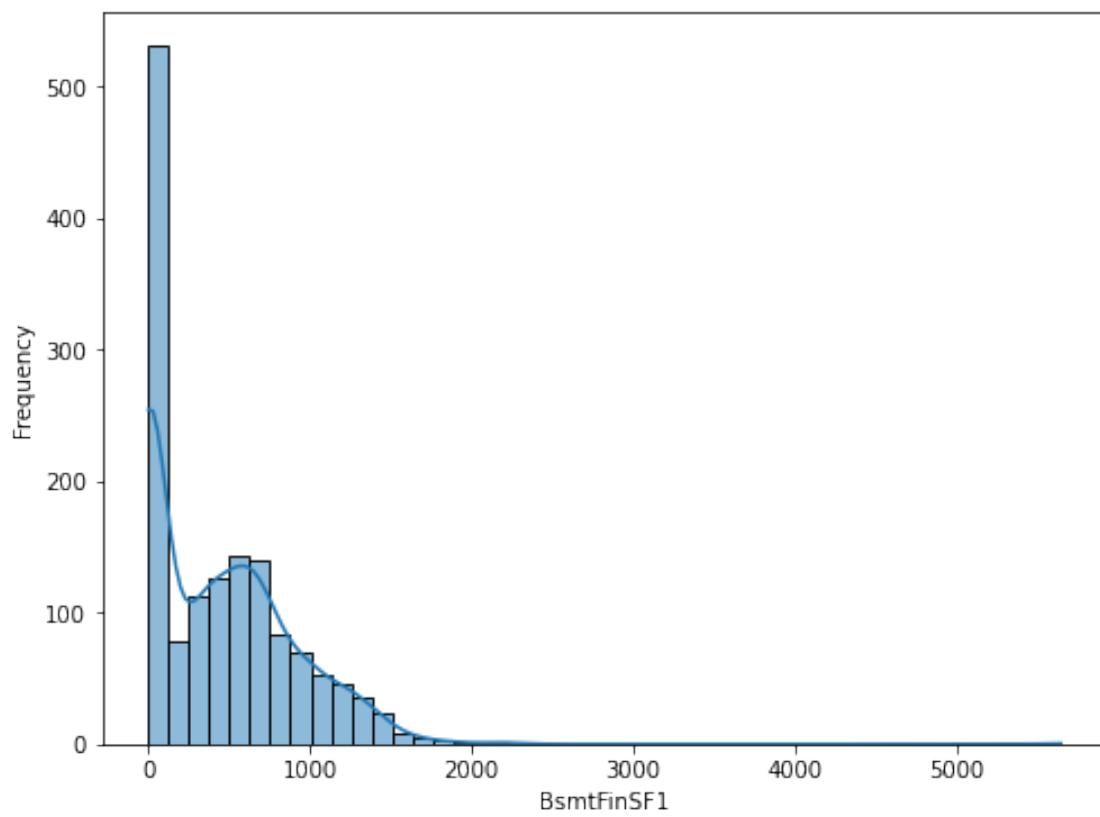




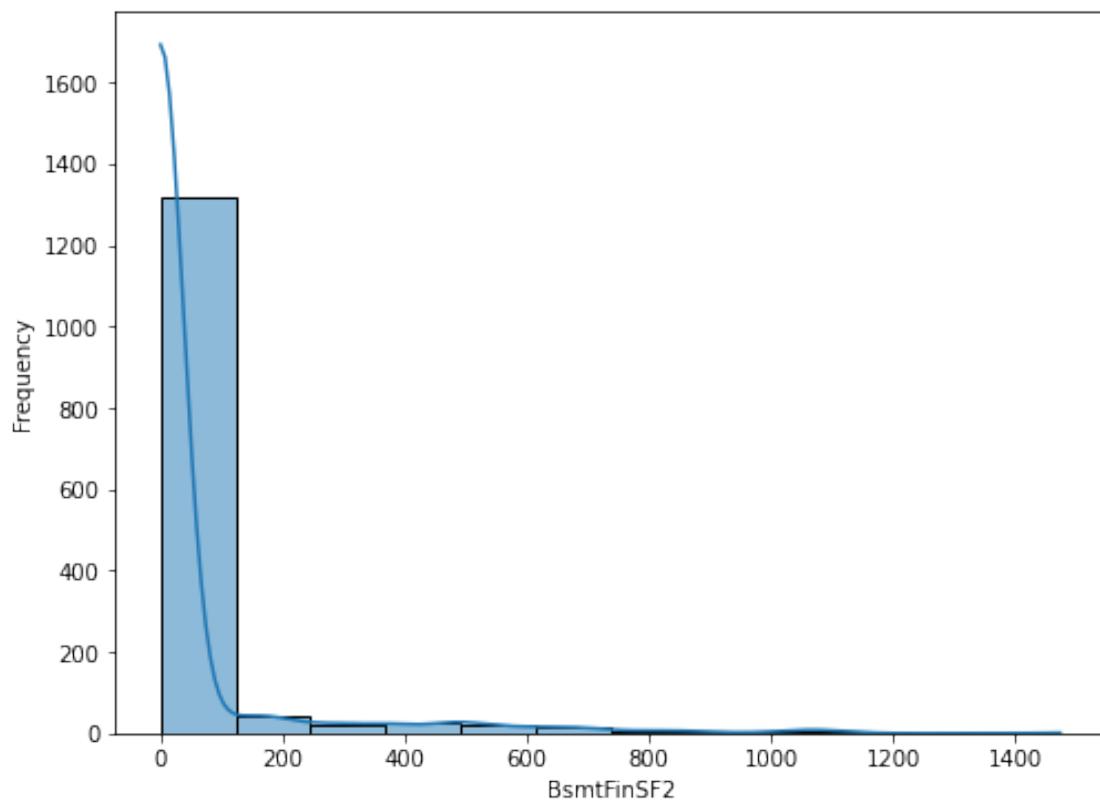
Distribution of MasVnrArea



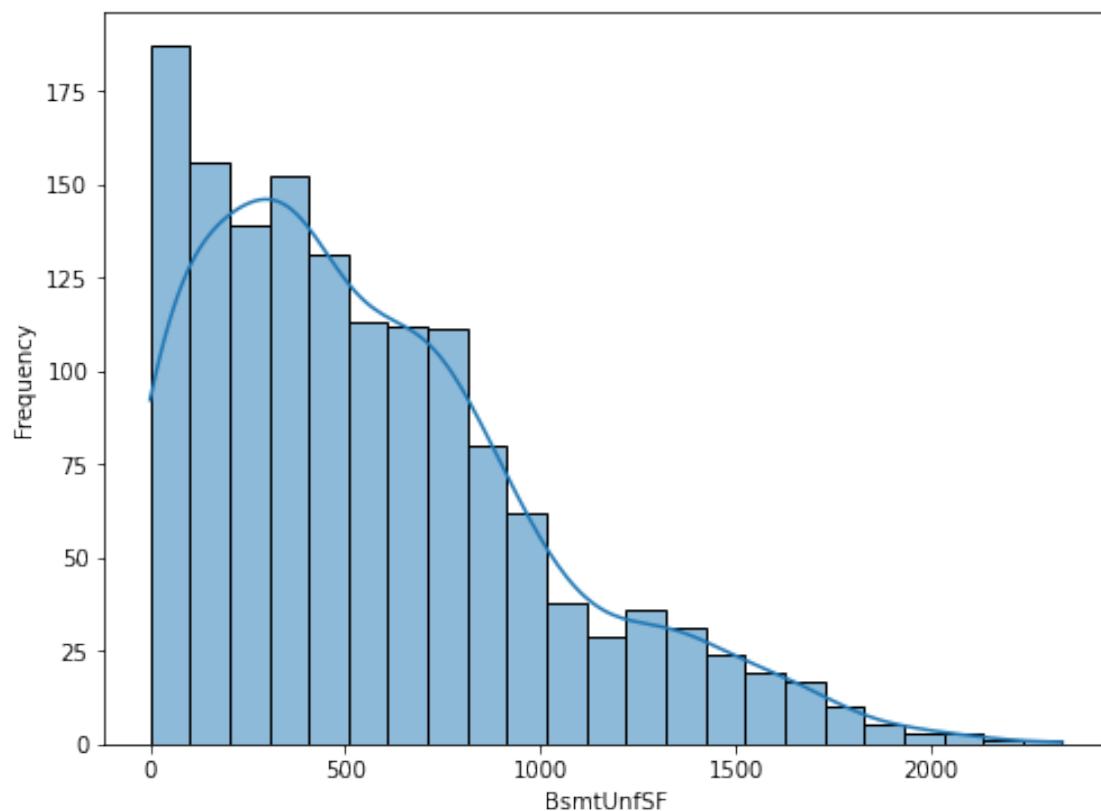
Distribution of BsmtFinSF1



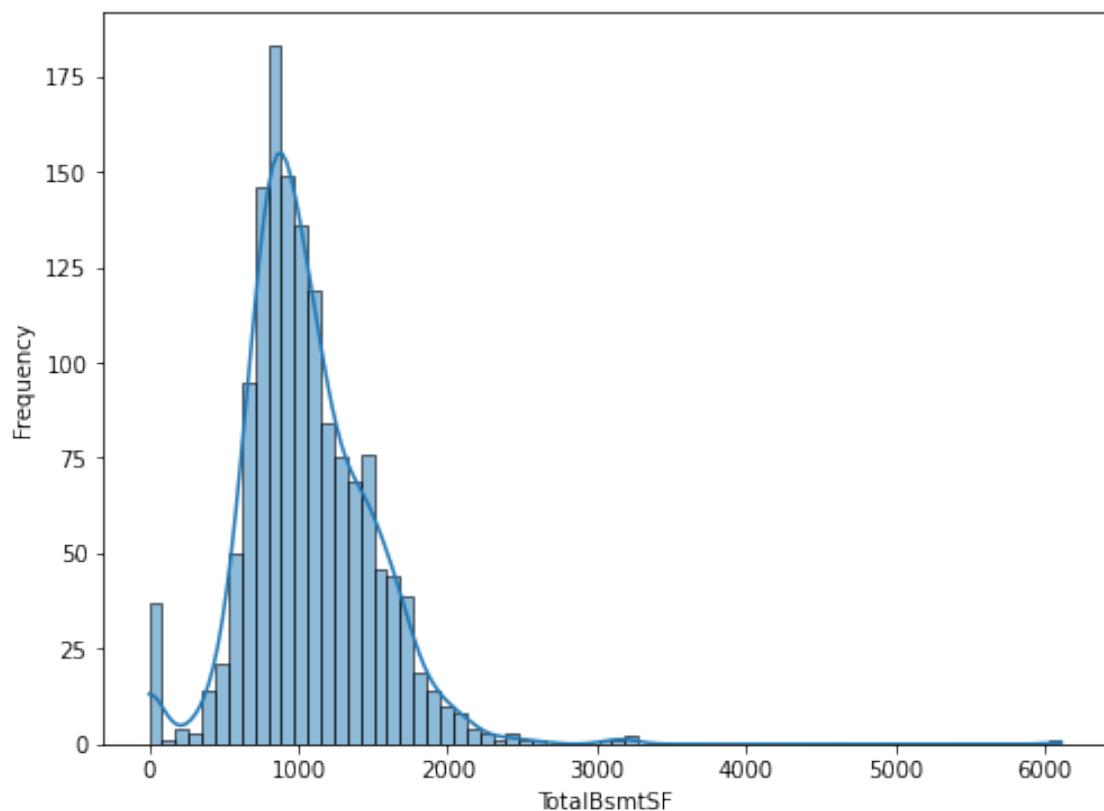
Distribution of BsmtFinSF2



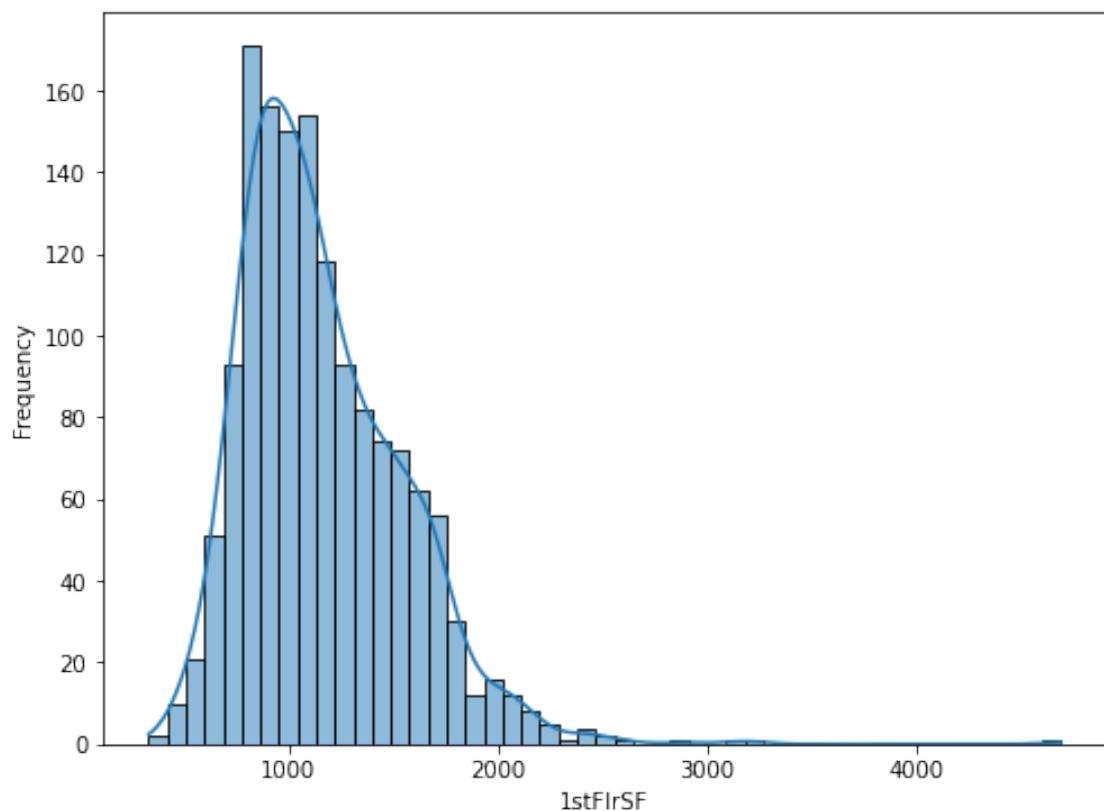
Distribution of BsmtUnfSF



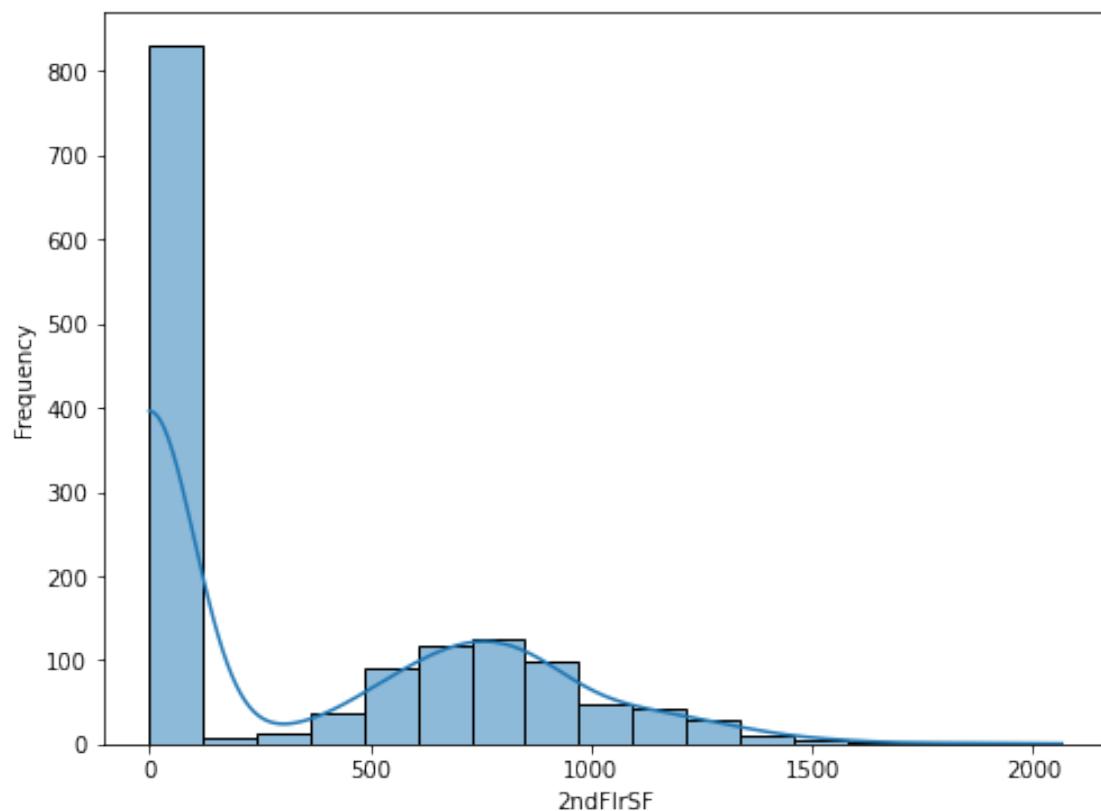
Distribution of TotalBsmtSF

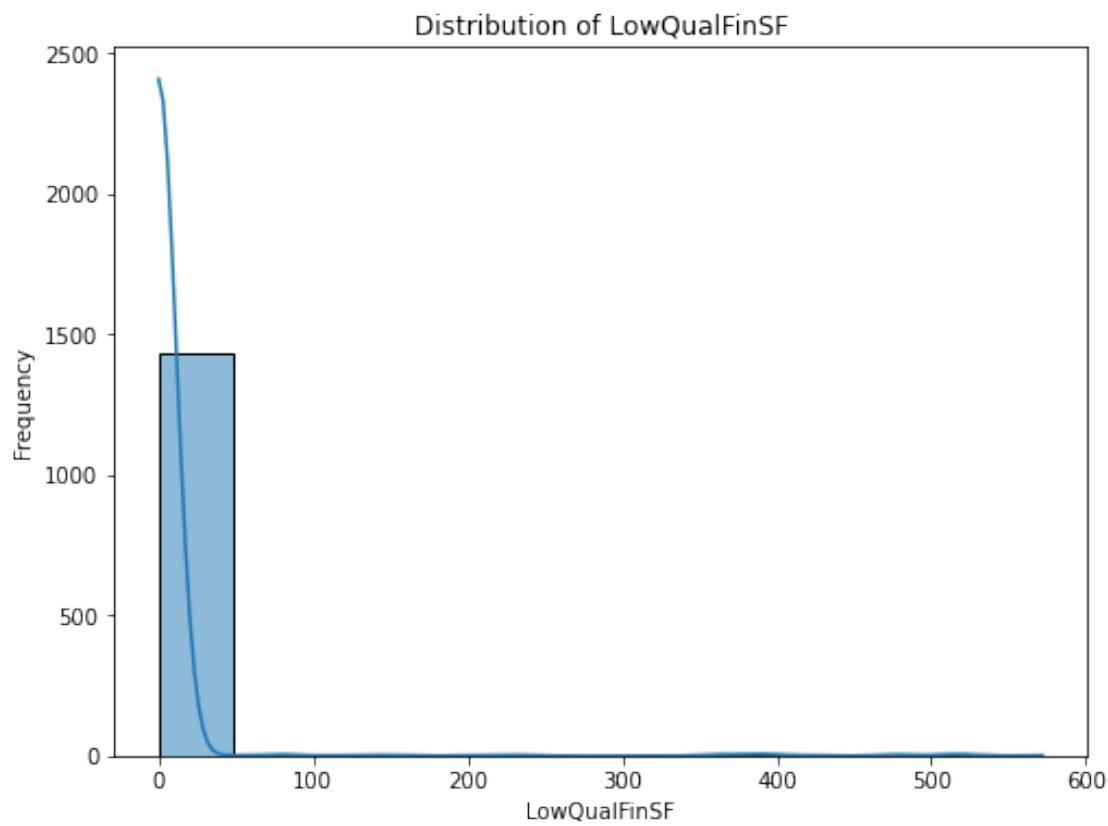


Distribution of 1stFlrSF

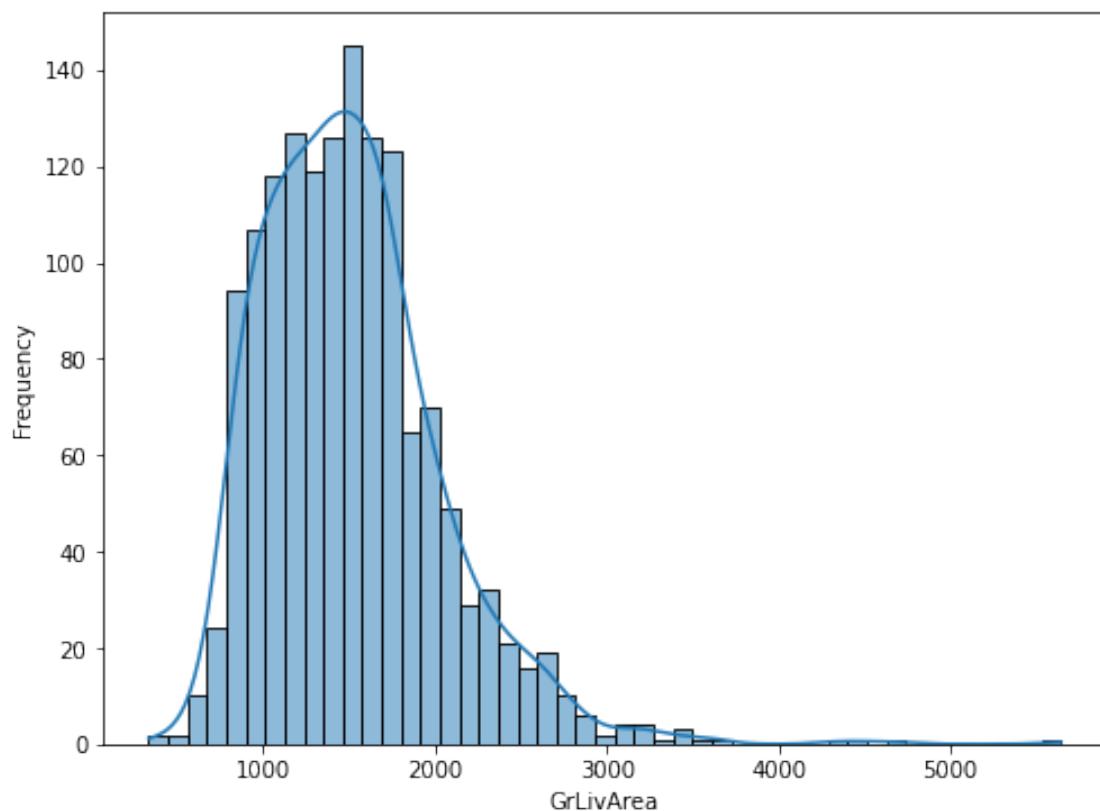


Distribution of 2ndFlrSF

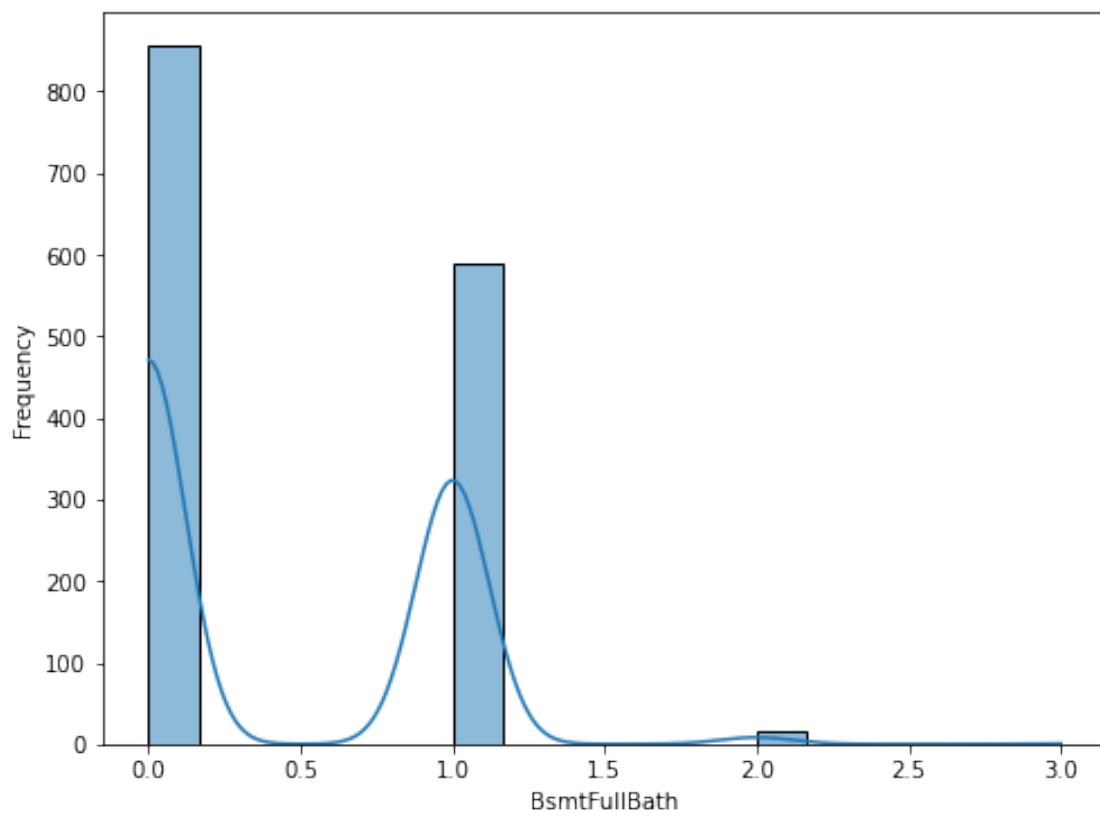




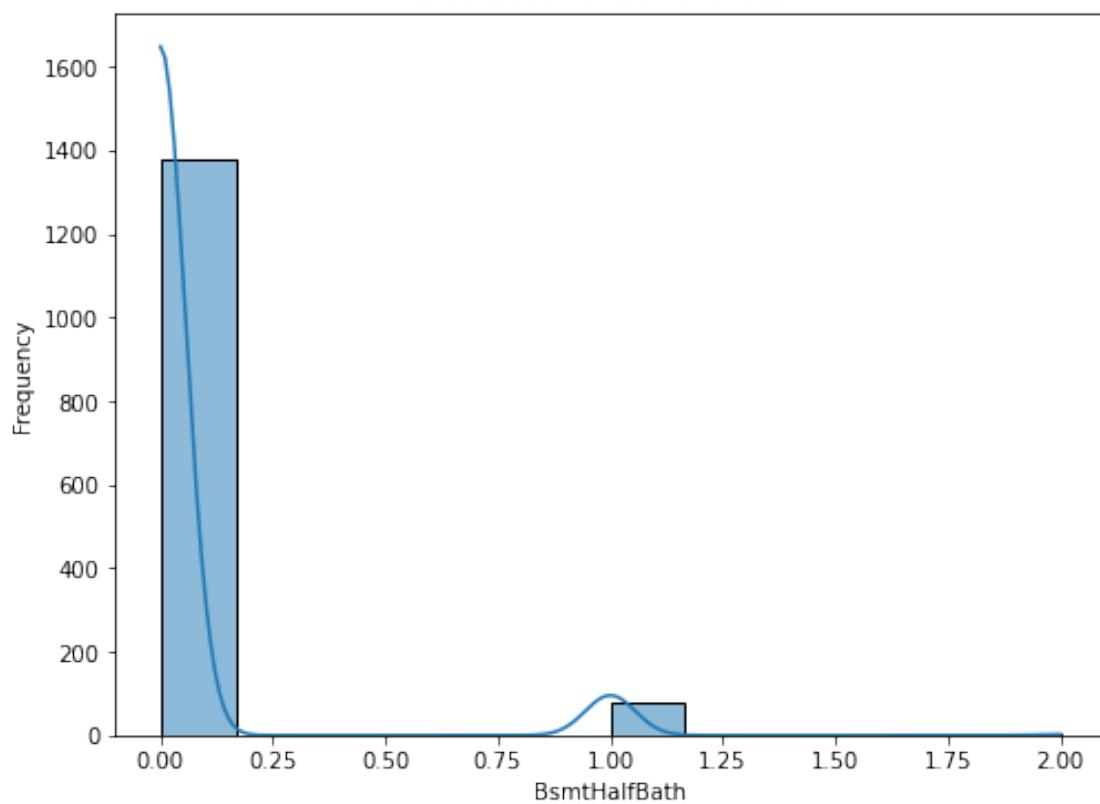
Distribution of GrLivArea

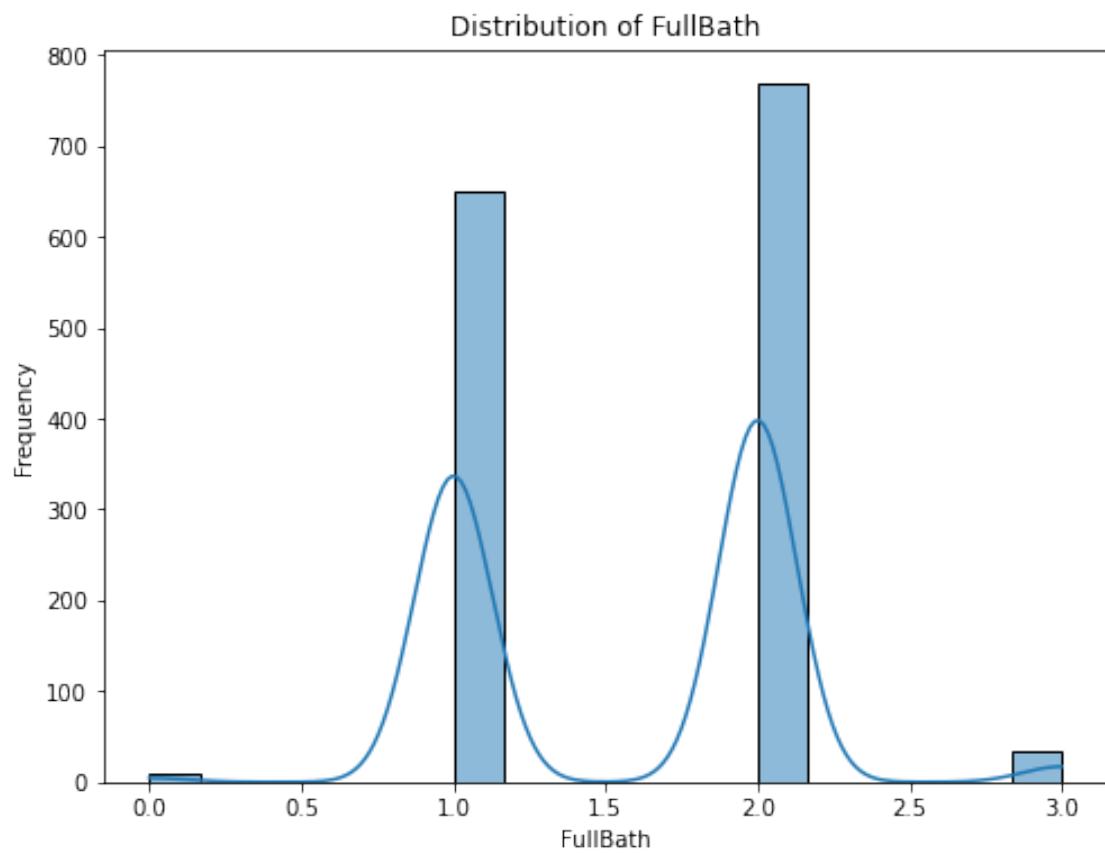


Distribution of BsmtFullBath

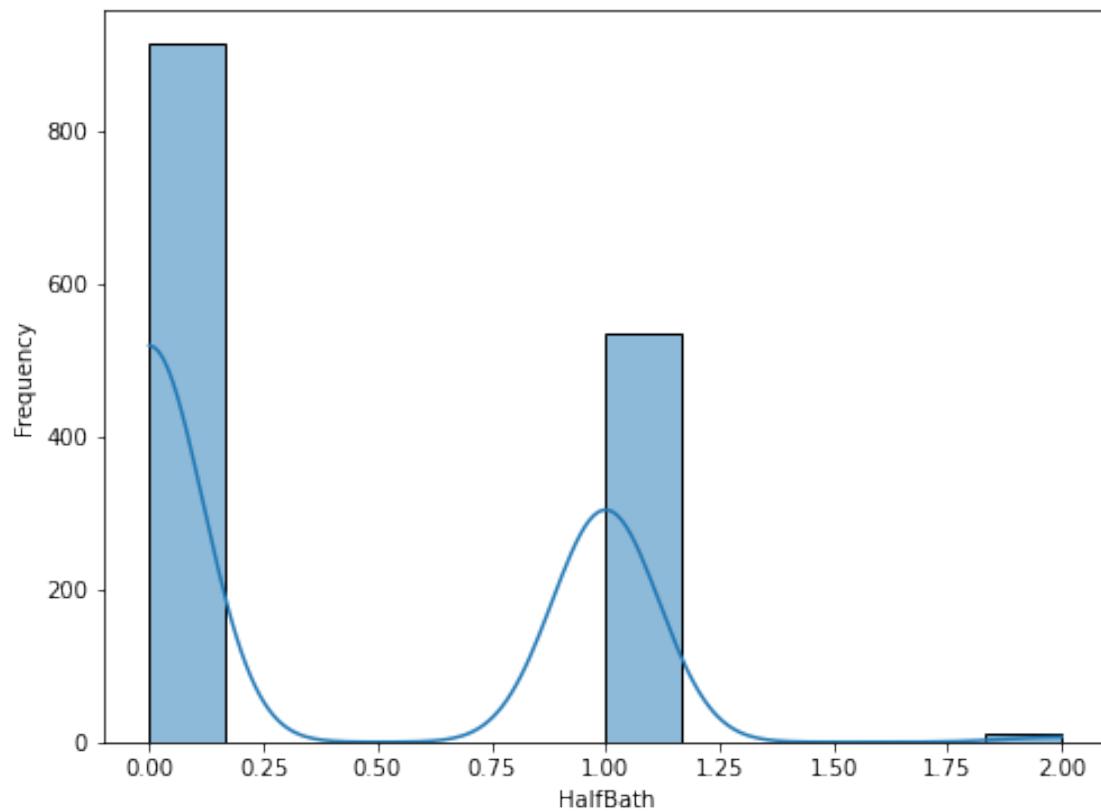


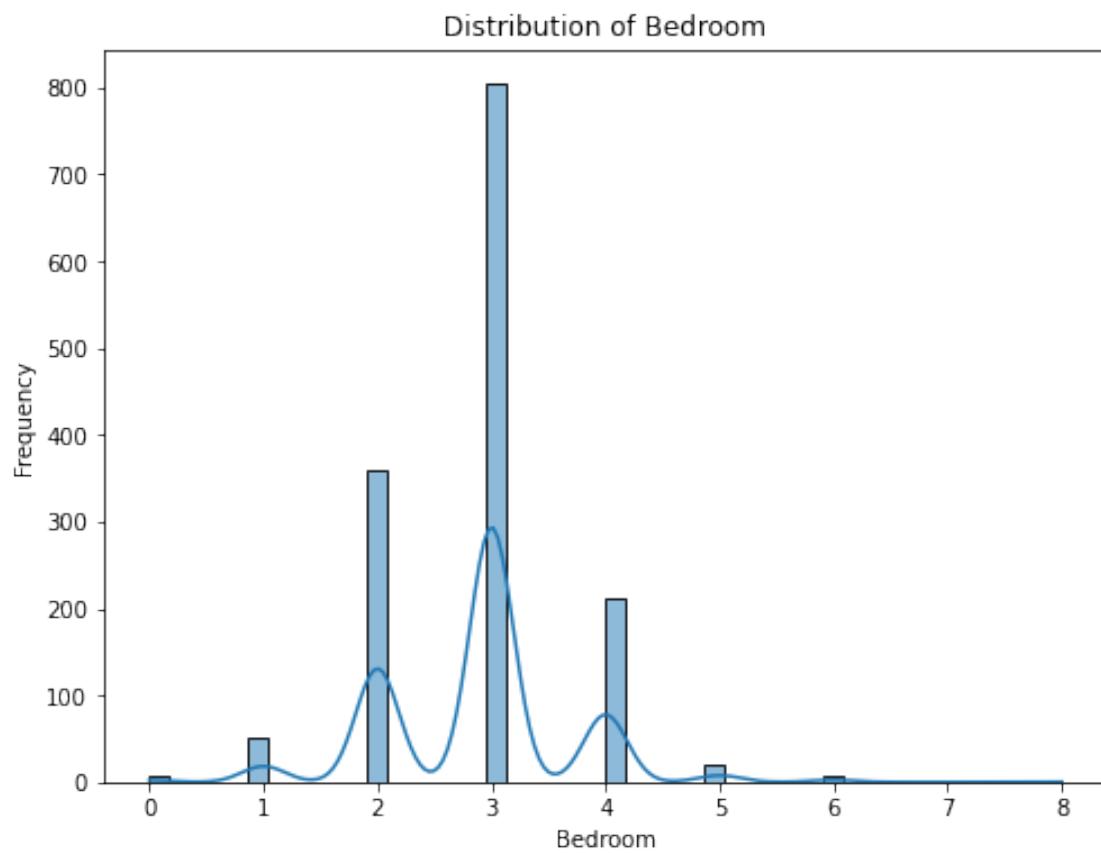
Distribution of BsmtHalfBath



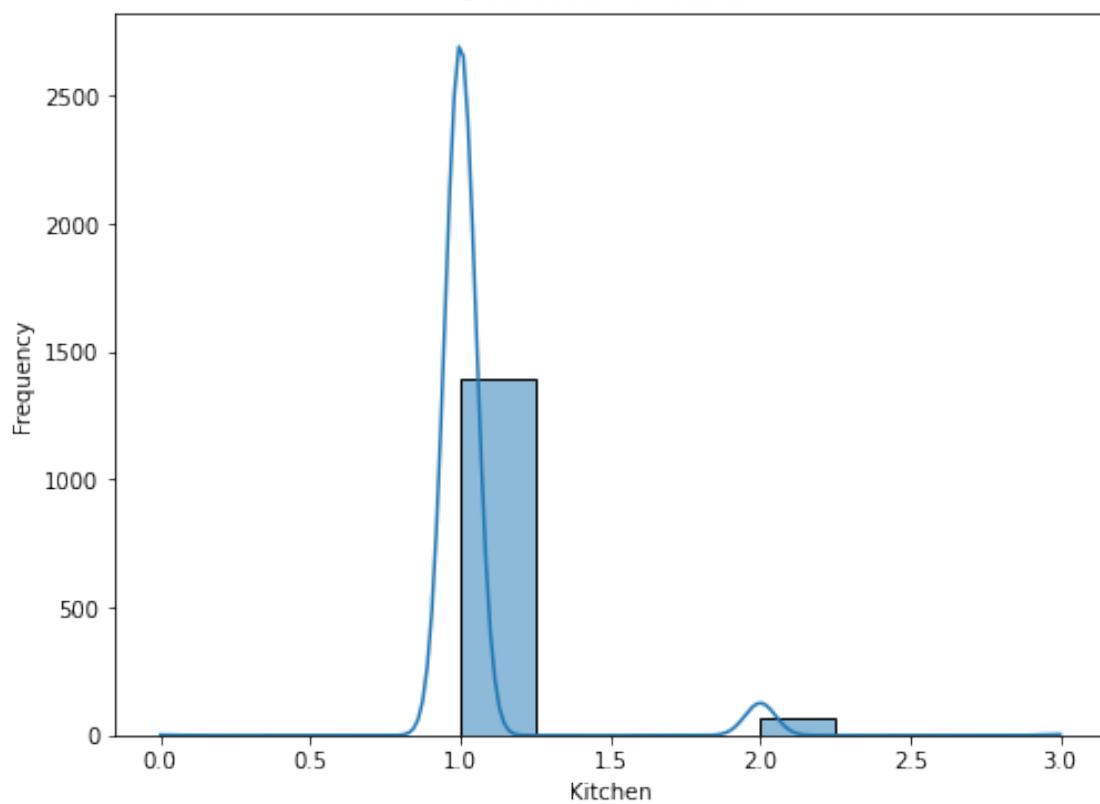


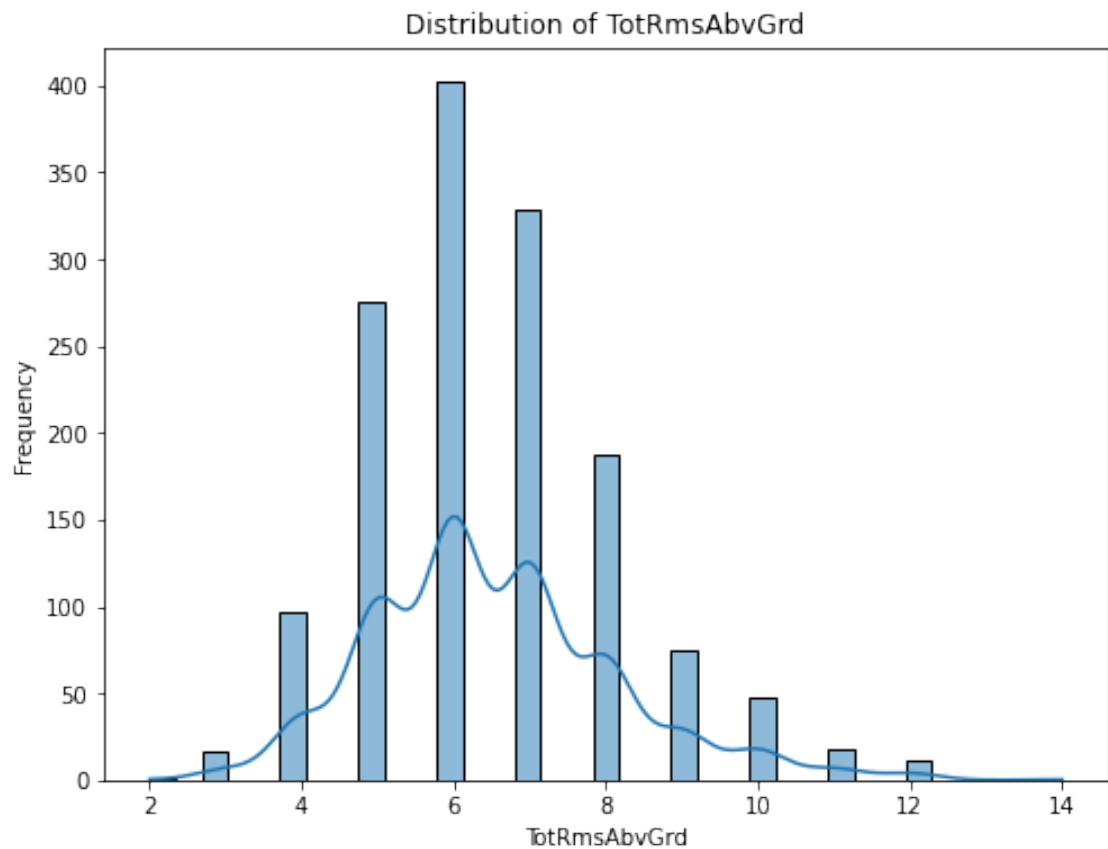
Distribution of HalfBath



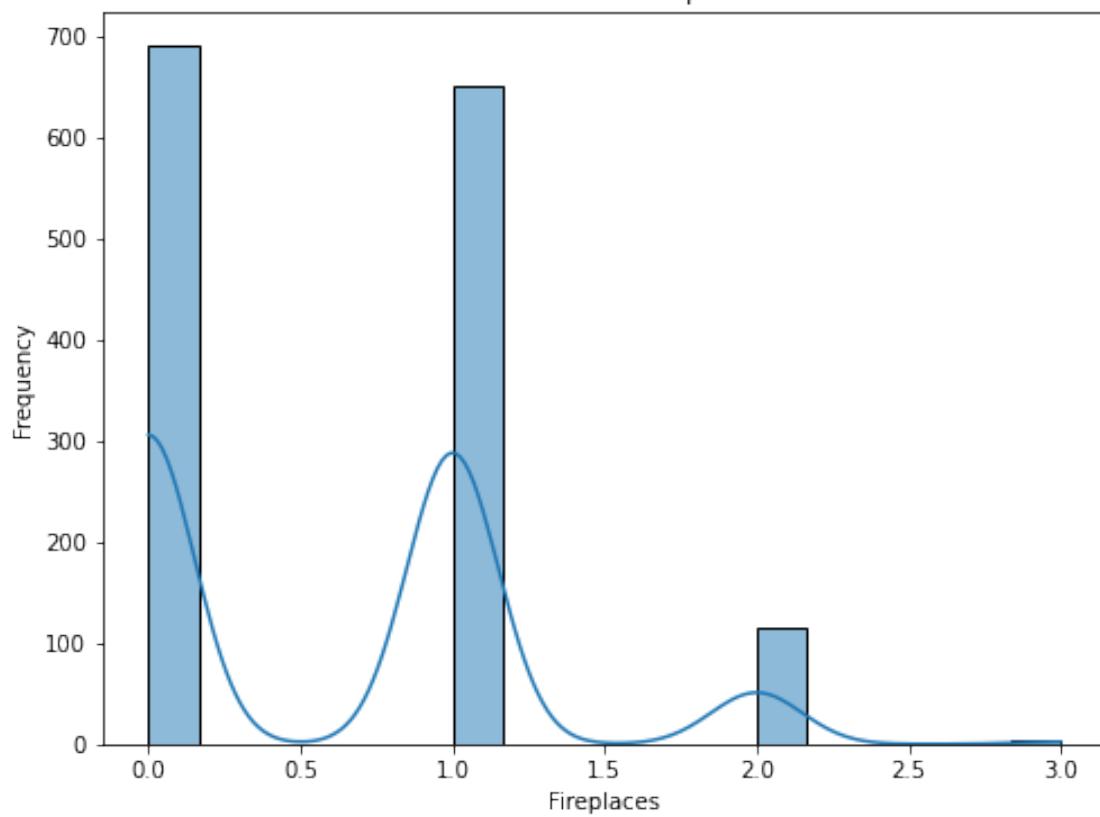


Distribution of Kitchen

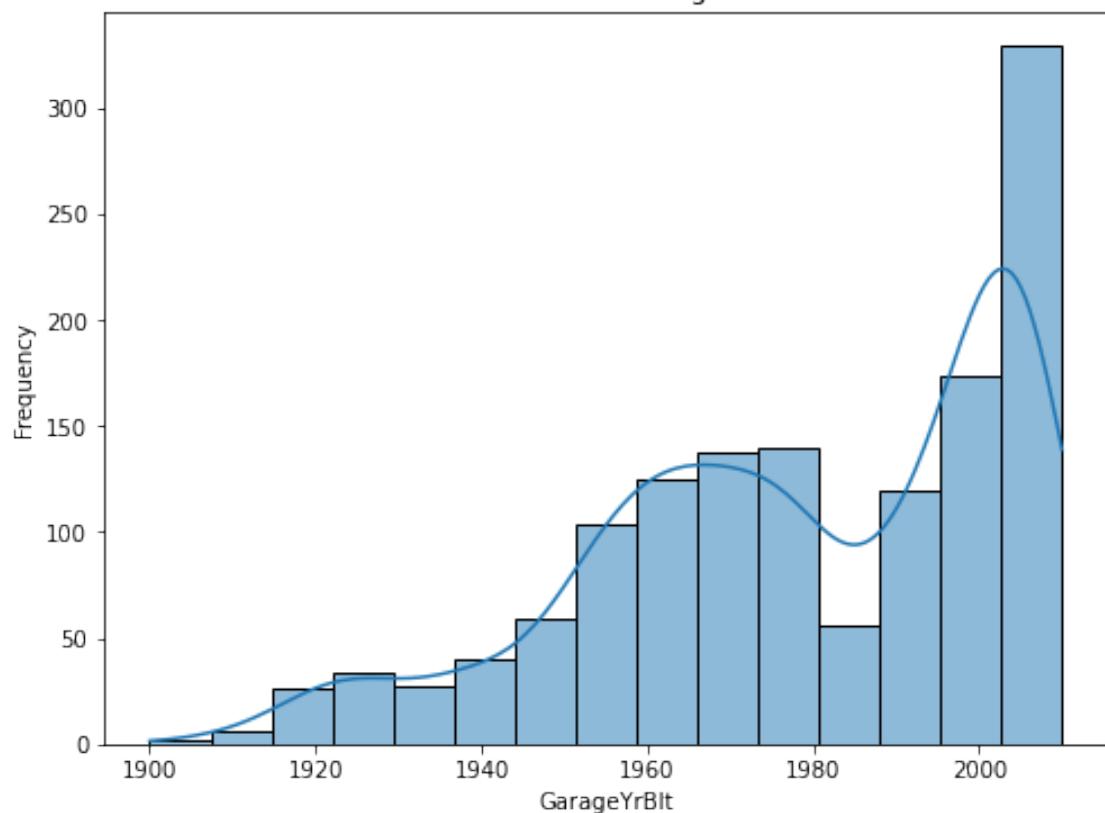




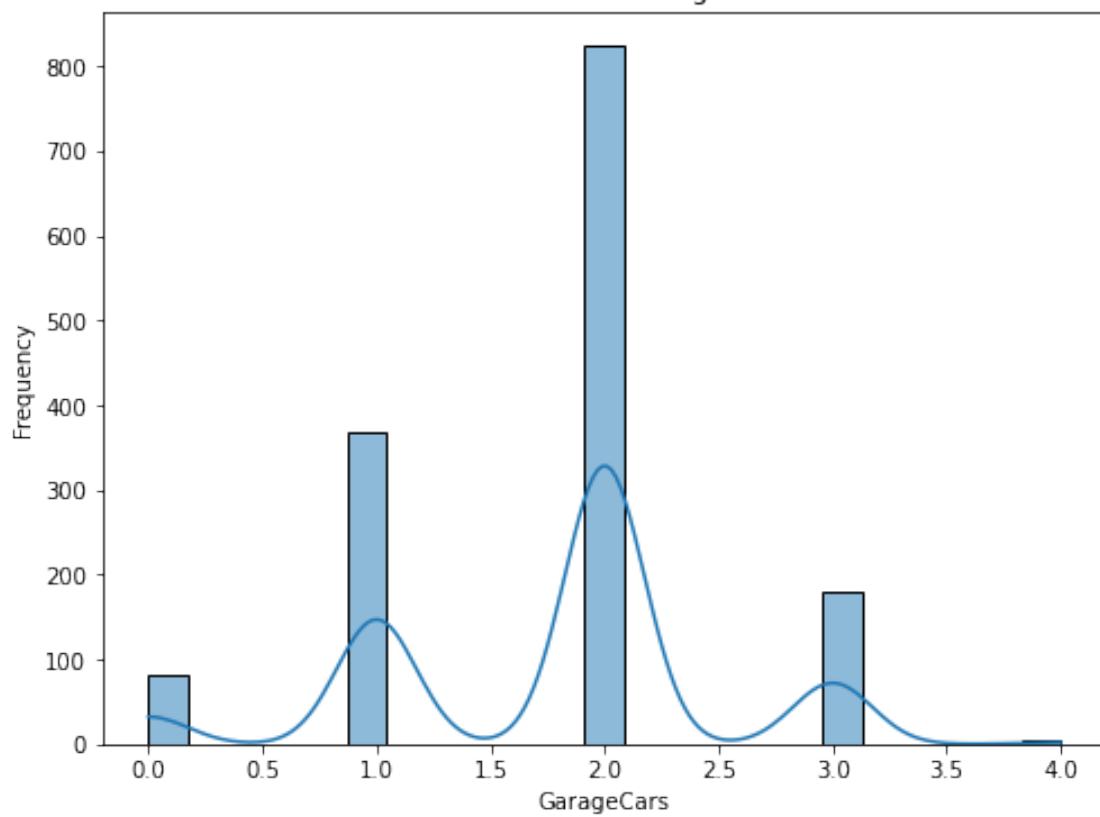
Distribution of Fireplaces



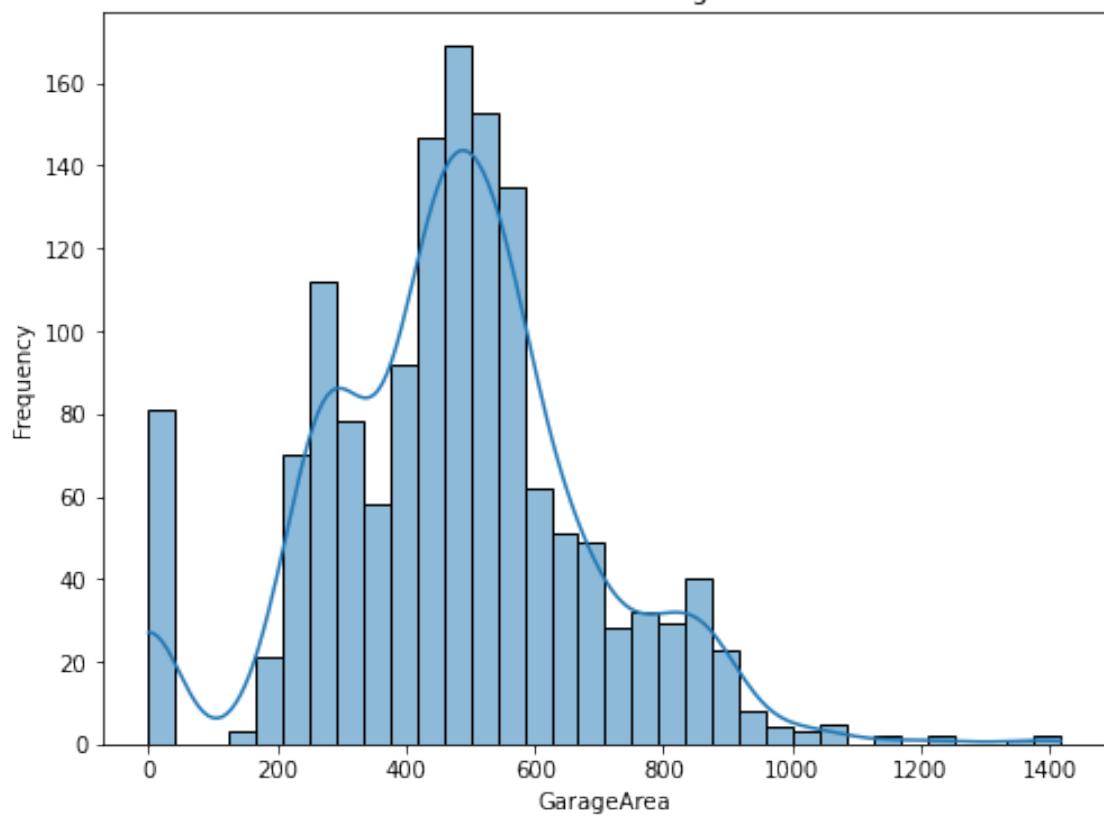
Distribution of GarageYrBlt

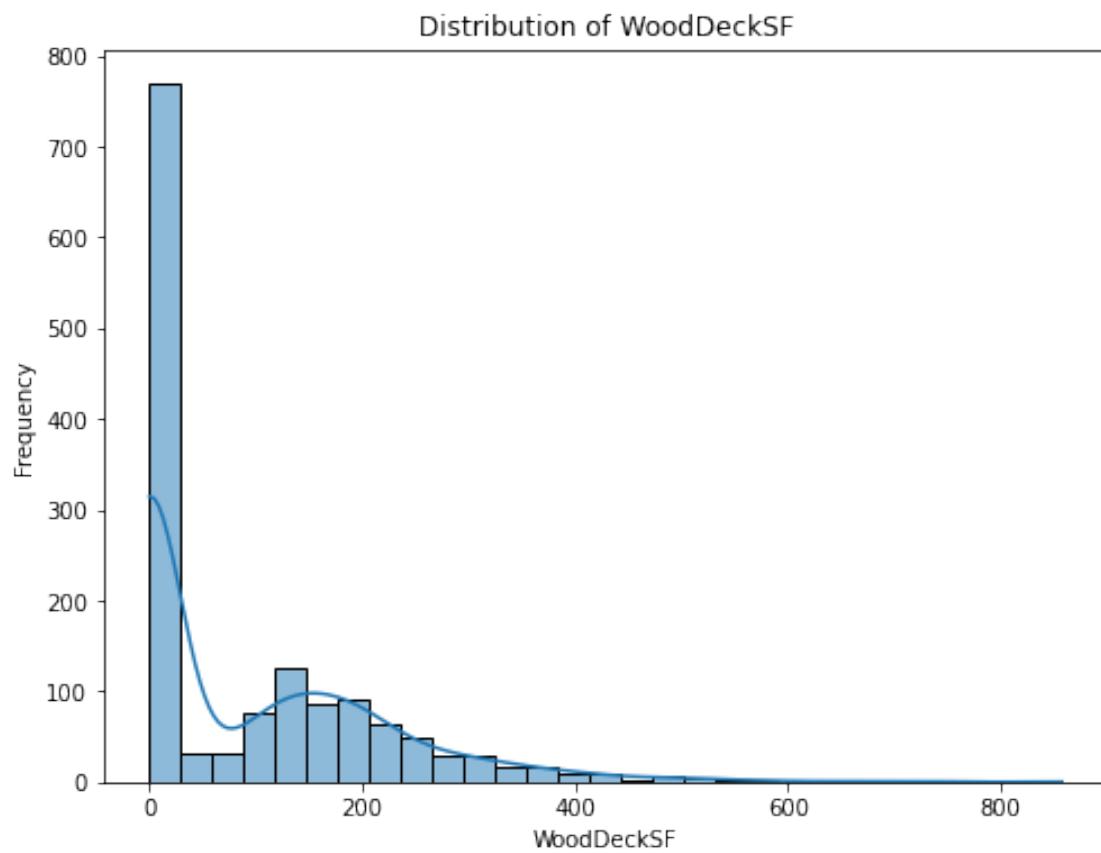


Distribution of GarageCars

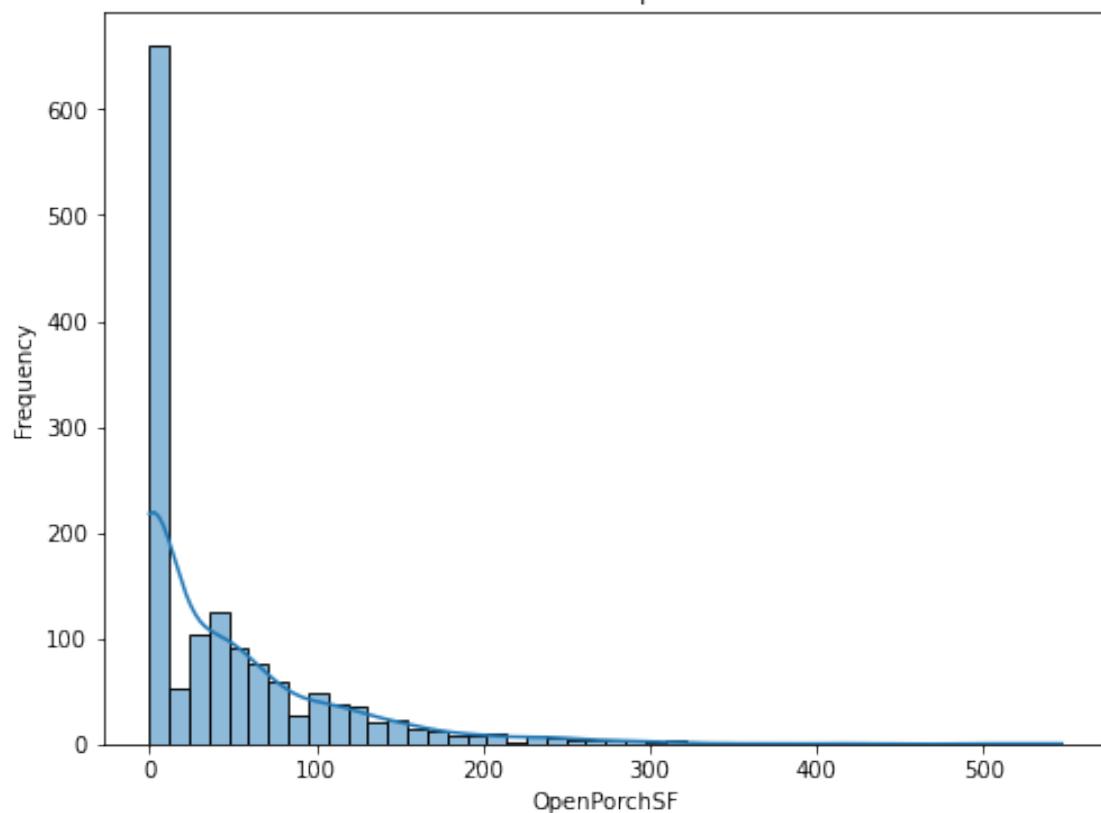


Distribution of GarageArea

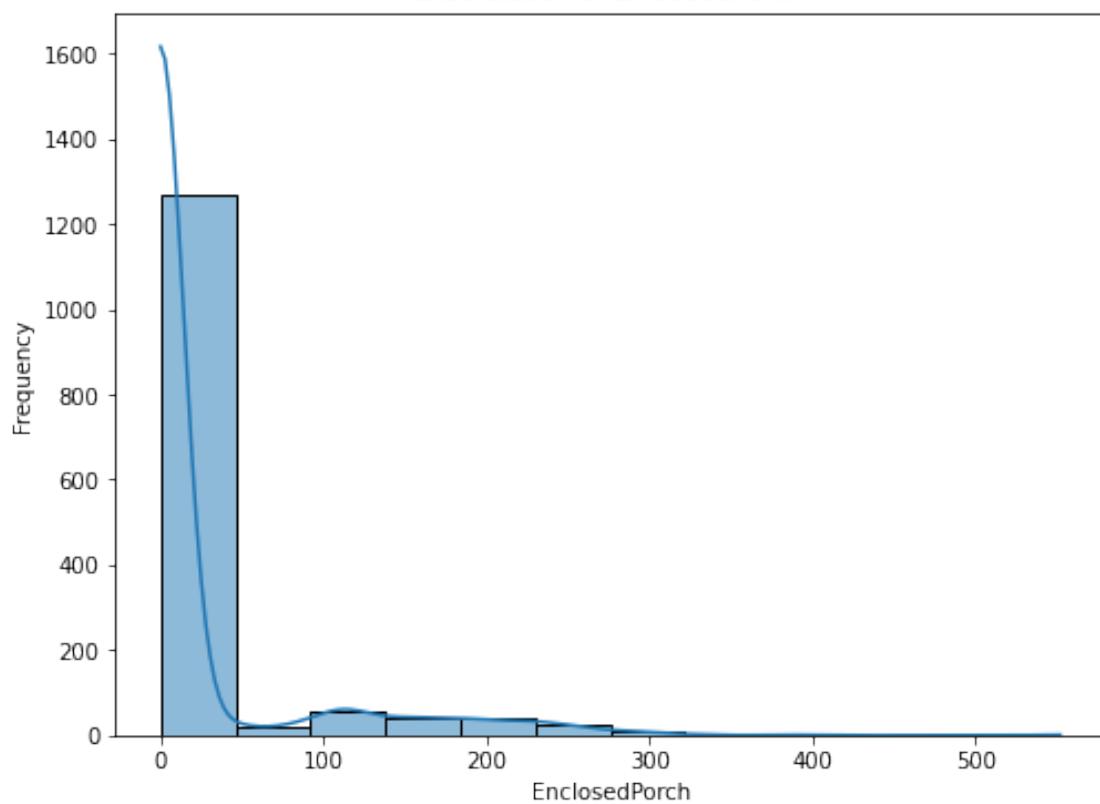




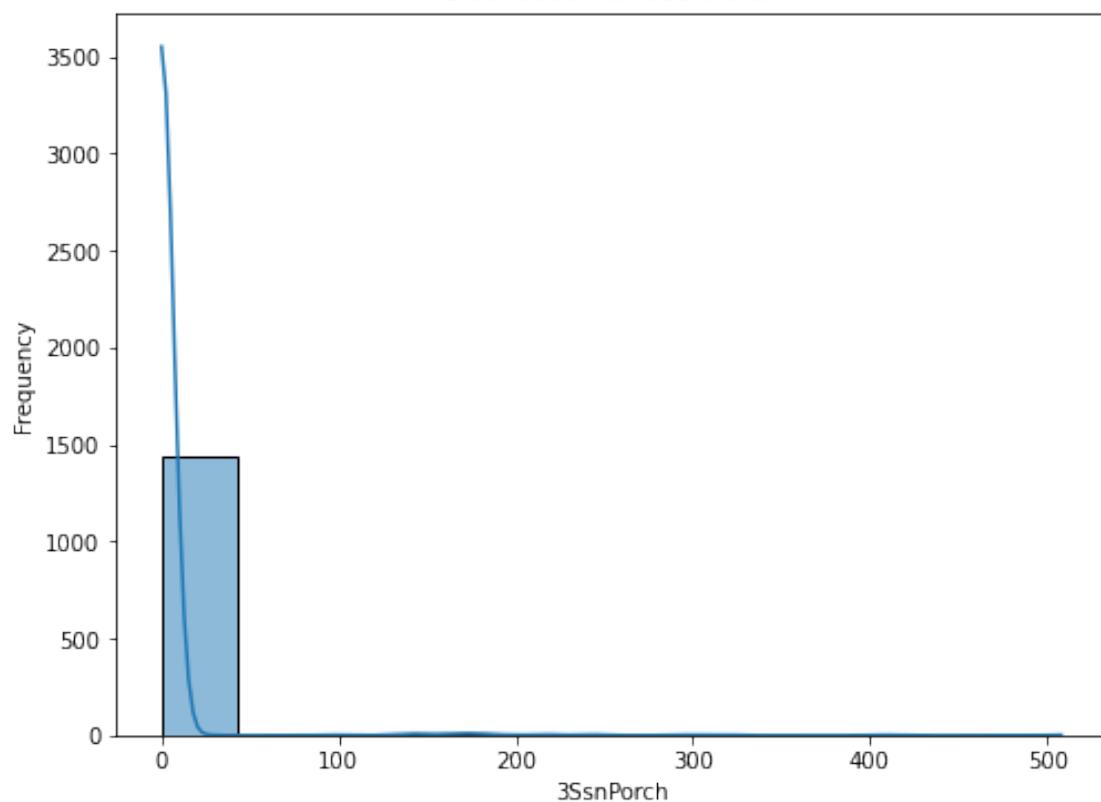
Distribution of OpenPorchSF



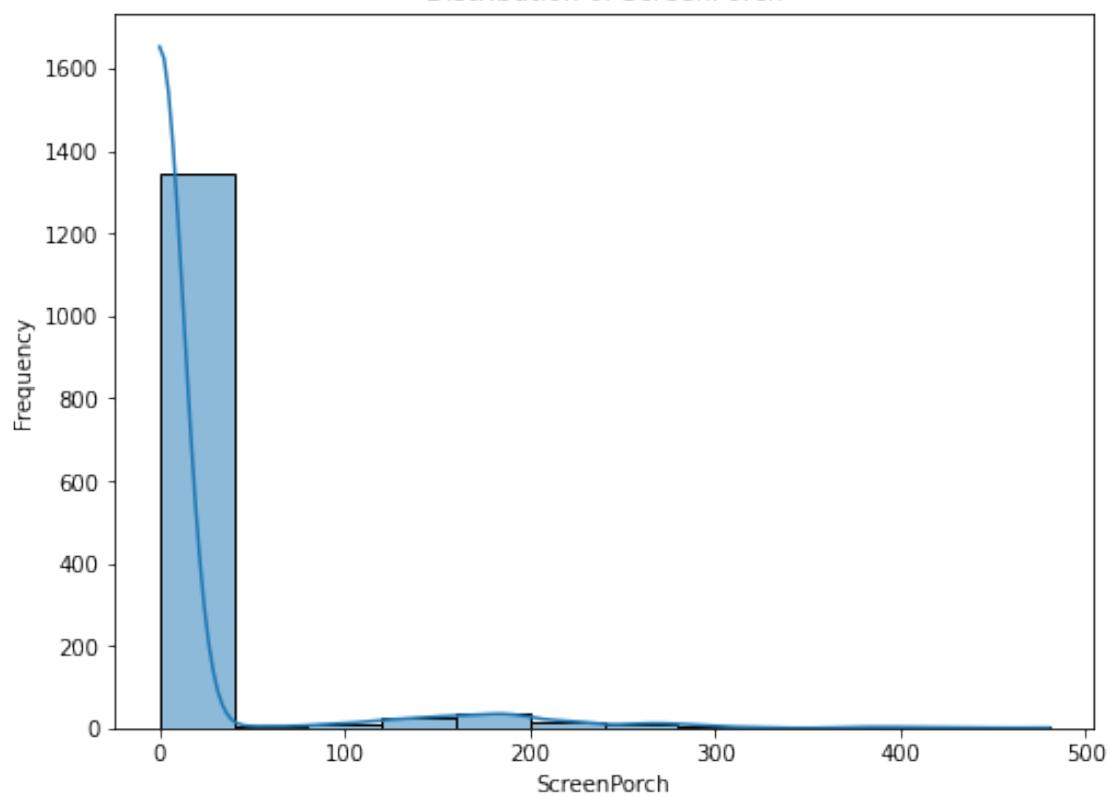
Distribution of EnclosedPorch

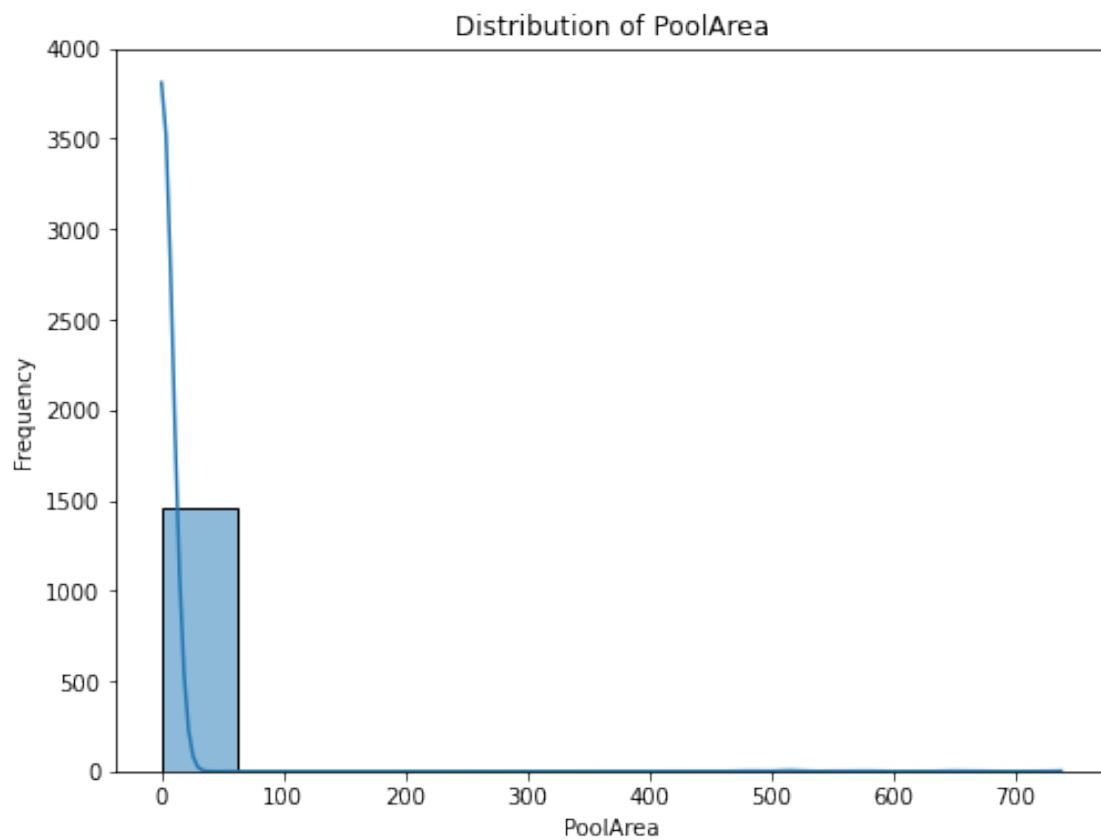


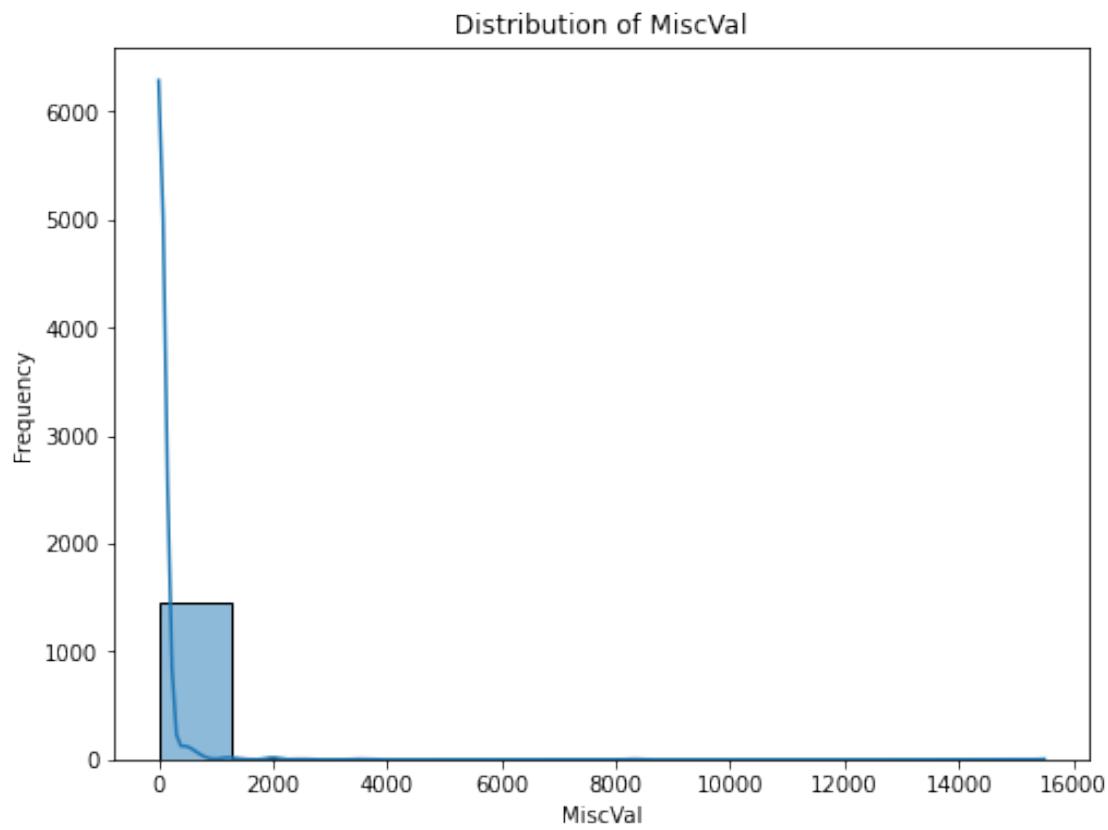
Distribution of 3SsnPorch



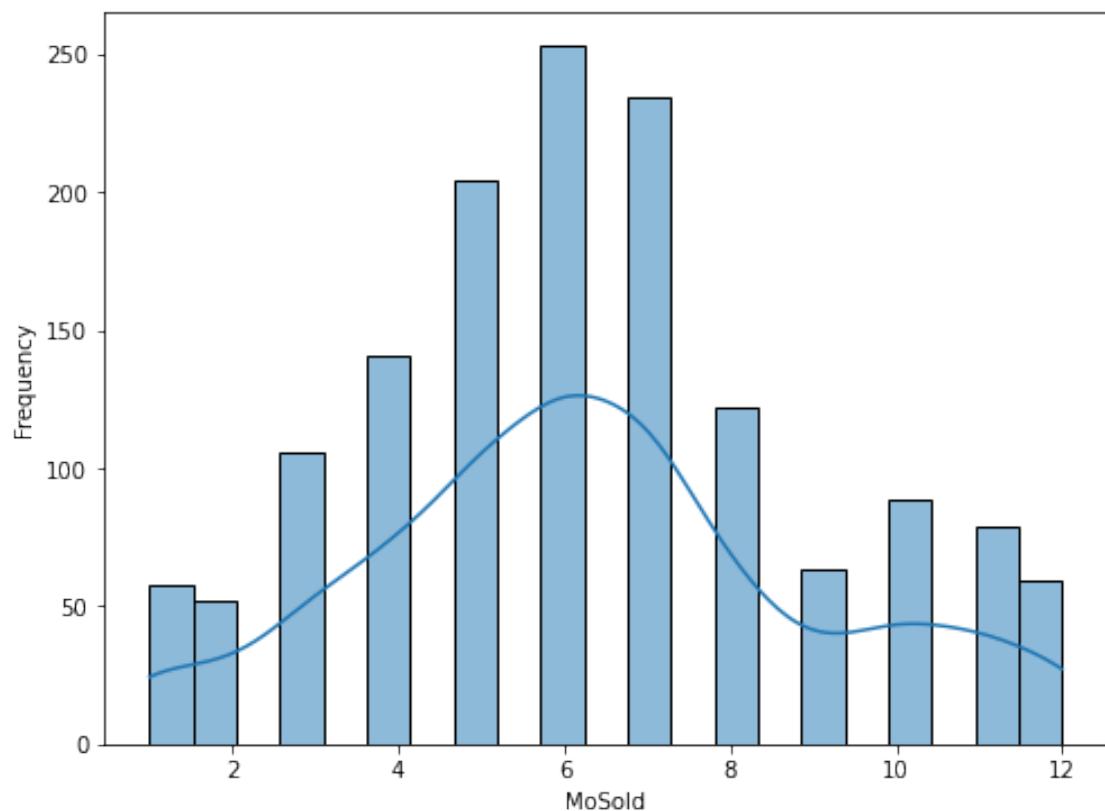
Distribution of ScreenPorch

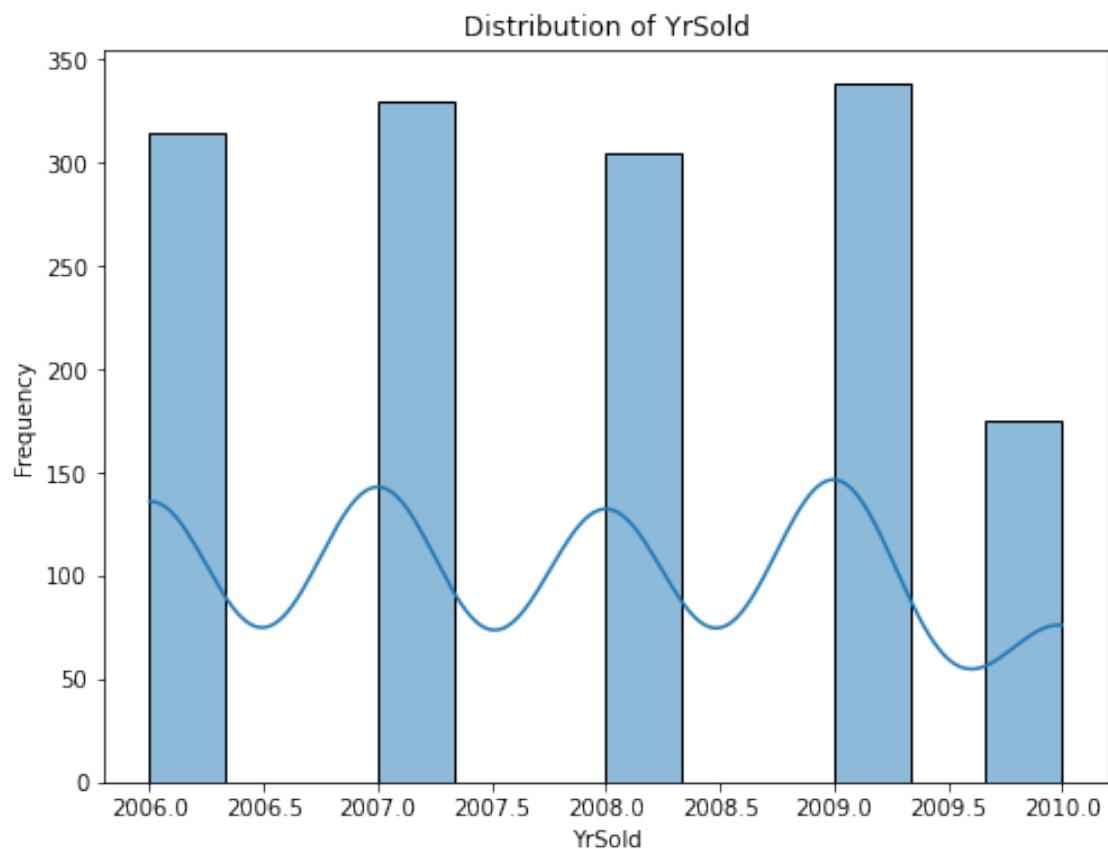


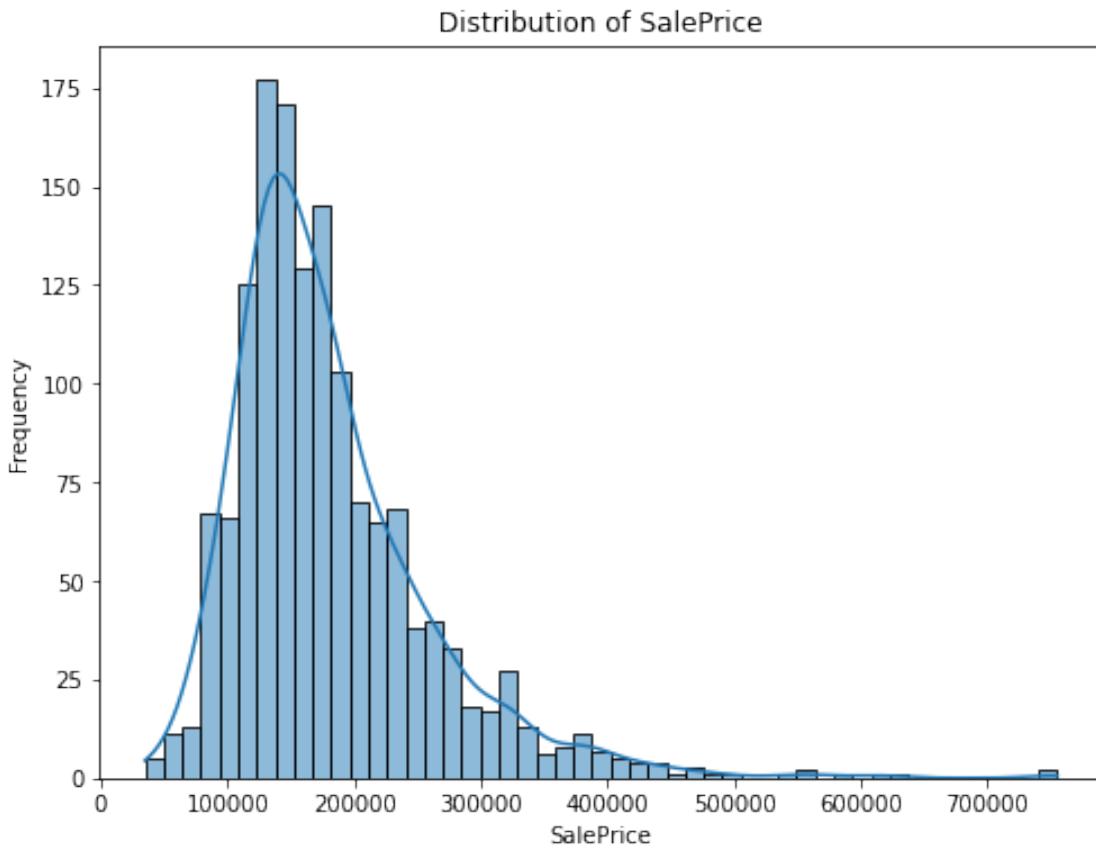




Distribution of MoSold







```
[24]: correlation_matrix = dataframe.corr()
significant_variables = correlation_matrix[abs(correlation_matrix) > 0.5].
    dropna(how='all', axis=1).dropna(how='all', axis=0)
print("Significant variables based on correlation matrix:")
print(significant_variables)
```

Significant variables based on correlation matrix:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	\
Id	1.0	NaN	NaN	NaN	NaN	NaN
MSSubClass	NaN	1.0	NaN	NaN	NaN	NaN
LotFrontage	NaN	NaN	1.0	NaN	NaN	NaN
LotArea	NaN	NaN	NaN	1.0	NaN	NaN
OverallQual	NaN	NaN	NaN	NaN	1.000000	
OverallCond	NaN	NaN	NaN	NaN	NaN	NaN
YearBuilt	NaN	NaN	NaN	NaN	0.572323	
YearRemodAdd	NaN	NaN	NaN	NaN	0.550684	
MasVnrArea	NaN	NaN	NaN	NaN	NaN	NaN
BsmtFinSF1	NaN	NaN	NaN	NaN	NaN	NaN
BsmtFinSF2	NaN	NaN	NaN	NaN	NaN	NaN
BsmtUnfSF	NaN	NaN	NaN	NaN	NaN	NaN

TotalBsmtSF	NaN	NaN	NaN	NaN	0.537808
1stFlrSF	NaN	NaN	NaN	NaN	NaN
2ndFlrSF	NaN	NaN	NaN	NaN	NaN
LowQualFinSF	NaN	NaN	NaN	NaN	NaN
GrLivArea	NaN	NaN	NaN	NaN	0.593007
BsmtFullBath	NaN	NaN	NaN	NaN	NaN
BsmtHalfBath	NaN	NaN	NaN	NaN	NaN
FullBath	NaN	NaN	NaN	NaN	0.550600
HalfBath	NaN	NaN	NaN	NaN	NaN
Bedroom	NaN	NaN	NaN	NaN	NaN
Kitchen	NaN	NaN	NaN	NaN	NaN
TotRmsAbvGrd	NaN	NaN	NaN	NaN	NaN
Fireplaces	NaN	NaN	NaN	NaN	NaN
GarageYrBlt	NaN	NaN	NaN	NaN	0.518018
GarageCars	NaN	NaN	NaN	NaN	0.600671
GarageArea	NaN	NaN	NaN	NaN	0.562022
WoodDeckSF	NaN	NaN	NaN	NaN	NaN
OpenPorchSF	NaN	NaN	NaN	NaN	NaN
EnclosedPorch	NaN	NaN	NaN	NaN	NaN
3SsnPorch	NaN	NaN	NaN	NaN	NaN
ScreenPorch	NaN	NaN	NaN	NaN	NaN
PoolArea	NaN	NaN	NaN	NaN	NaN
MiscVal	NaN	NaN	NaN	NaN	NaN
MoSold	NaN	NaN	NaN	NaN	NaN
YrSold	NaN	NaN	NaN	NaN	NaN
SalePrice	NaN	NaN	NaN	NaN	0.790982

	OverallCond	YearBuilt	YearRemodAdd	MasVnrArea	BsmtFinSF1	\
Id	NaN	NaN	NaN	NaN	NaN	NaN
MSSubClass	NaN	NaN	NaN	NaN	NaN	NaN
LotFrontage	NaN	NaN	NaN	NaN	NaN	NaN
LotArea	NaN	NaN	NaN	NaN	NaN	NaN
OverallQual	NaN	0.572323	0.550684	NaN	NaN	NaN
OverallCond	1.0	NaN	NaN	NaN	NaN	NaN
YearBuilt	NaN	1.000000	0.592855	NaN	NaN	NaN
YearRemodAdd	NaN	0.592855	1.000000	NaN	NaN	NaN
MasVnrArea	NaN	NaN	NaN	1.0	NaN	
BsmtFinSF1	NaN	NaN	NaN	NaN	1.000000	
BsmtFinSF2	NaN	NaN	NaN	NaN	NaN	NaN
BsmtUnfSF	NaN	NaN	NaN	NaN	NaN	NaN
TotalBsmtSF	NaN	NaN	NaN	NaN	0.522396	
1stFlrSF	NaN	NaN	NaN	NaN	NaN	NaN
2ndFlrSF	NaN	NaN	NaN	NaN	NaN	NaN
LowQualFinSF	NaN	NaN	NaN	NaN	NaN	NaN
GrLivArea	NaN	NaN	NaN	NaN	NaN	NaN
BsmtFullBath	NaN	NaN	NaN	NaN	0.649212	
BsmtHalfBath	NaN	NaN	NaN	NaN	NaN	NaN
FullBath	NaN	NaN	NaN	NaN	NaN	NaN

HalfBath	NaN	NaN	NaN	NaN	NaN
Bedroom	NaN	NaN	NaN	NaN	NaN
Kitchen	NaN	NaN	NaN	NaN	NaN
TotRmsAbvGrd	NaN	NaN	NaN	NaN	NaN
Fireplaces	NaN	NaN	NaN	NaN	NaN
GarageYrBlt	NaN	0.780555	0.618130	NaN	NaN
GarageCars	NaN	0.537850	NaN	NaN	NaN
GarageArea	NaN	NaN	NaN	NaN	NaN
WoodDeckSF	NaN	NaN	NaN	NaN	NaN
OpenPorchSF	NaN	NaN	NaN	NaN	NaN
EnclosedPorch	NaN	NaN	NaN	NaN	NaN
3SsnPorch	NaN	NaN	NaN	NaN	NaN
ScreenPorch	NaN	NaN	NaN	NaN	NaN
PoolArea	NaN	NaN	NaN	NaN	NaN
MiscVal	NaN	NaN	NaN	NaN	NaN
MoSold	NaN	NaN	NaN	NaN	NaN
YrSold	NaN	NaN	NaN	NaN	NaN
SalePrice	NaN	0.522897	0.507101	NaN	NaN

	...	WoodDeckSF	OpenPorchSF	EnclosedPorch	3SsnPorch	\
Id	...	NaN	NaN	NaN	NaN	
MSSubClass	...	NaN	NaN	NaN	NaN	
LotFrontage	...	NaN	NaN	NaN	NaN	
LotArea	...	NaN	NaN	NaN	NaN	
OverallQual	...	NaN	NaN	NaN	NaN	
OverallCond	...	NaN	NaN	NaN	NaN	
YearBuilt	...	NaN	NaN	NaN	NaN	
YearRemodAdd	...	NaN	NaN	NaN	NaN	
MasVnrArea	...	NaN	NaN	NaN	NaN	
BsmtFinSF1	...	NaN	NaN	NaN	NaN	
BsmtFinSF2	...	NaN	NaN	NaN	NaN	
BsmtUnfSF	...	NaN	NaN	NaN	NaN	
TotalBsmtSF	...	NaN	NaN	NaN	NaN	
1stFlrSF	...	NaN	NaN	NaN	NaN	
2ndFlrSF	...	NaN	NaN	NaN	NaN	
LowQualFinSF	...	NaN	NaN	NaN	NaN	
GrLivArea	...	NaN	NaN	NaN	NaN	
BsmtFullBath	...	NaN	NaN	NaN	NaN	
BsmtHalfBath	...	NaN	NaN	NaN	NaN	
FullBath	...	NaN	NaN	NaN	NaN	
HalfBath	...	NaN	NaN	NaN	NaN	
Bedroom	...	NaN	NaN	NaN	NaN	
Kitchen	...	NaN	NaN	NaN	NaN	
TotRmsAbvGrd	...	NaN	NaN	NaN	NaN	
Fireplaces	...	NaN	NaN	NaN	NaN	
GarageYrBlt	...	NaN	NaN	NaN	NaN	
GarageCars	...	NaN	NaN	NaN	NaN	
GarageArea	...	NaN	NaN	NaN	NaN	

WoodDeckSF	...	1.0	NaN	NaN	NaN	NaN
OpenPorchSF	...	NaN	1.0	NaN	NaN	NaN
EnclosedPorch	...	NaN	NaN	1.0	NaN	NaN
3SsnPorch	...	NaN	NaN	NaN	1.0	NaN
ScreenPorch	...	NaN	NaN	NaN	NaN	NaN
PoolArea	...	NaN	NaN	NaN	NaN	NaN
MiscVal	...	NaN	NaN	NaN	NaN	NaN
MoSold	...	NaN	NaN	NaN	NaN	NaN
YrSold	...	NaN	NaN	NaN	NaN	NaN
SalePrice	...	NaN	NaN	NaN	NaN	NaN

	ScreenPorch	PoolArea	MiscVal	MoSold	YrSold	SalePrice
Id	NaN	NaN	NaN	NaN	NaN	NaN
MSSubClass	NaN	NaN	NaN	NaN	NaN	NaN
LotFrontage	NaN	NaN	NaN	NaN	NaN	NaN
LotArea	NaN	NaN	NaN	NaN	NaN	NaN
OverallQual	NaN	NaN	NaN	NaN	NaN	0.790982
OverallCond	NaN	NaN	NaN	NaN	NaN	NaN
YearBuilt	NaN	NaN	NaN	NaN	NaN	0.522897
YearRemodAdd	NaN	NaN	NaN	NaN	NaN	0.507101
MasVnrArea	NaN	NaN	NaN	NaN	NaN	NaN
BsmtFinSF1	NaN	NaN	NaN	NaN	NaN	NaN
BsmtFinSF2	NaN	NaN	NaN	NaN	NaN	NaN
BsmtUnfSF	NaN	NaN	NaN	NaN	NaN	NaN
TotalBsmtSF	NaN	NaN	NaN	NaN	NaN	0.613581
1stFlrSF	NaN	NaN	NaN	NaN	NaN	0.605852
2ndFlrSF	NaN	NaN	NaN	NaN	NaN	NaN
LowQualFinSF	NaN	NaN	NaN	NaN	NaN	NaN
GrLivArea	NaN	NaN	NaN	NaN	NaN	0.708624
BsmtFullBath	NaN	NaN	NaN	NaN	NaN	NaN
BsmtHalfBath	NaN	NaN	NaN	NaN	NaN	NaN
FullBath	NaN	NaN	NaN	NaN	NaN	0.560664
HalfBath	NaN	NaN	NaN	NaN	NaN	NaN
Bedroom	NaN	NaN	NaN	NaN	NaN	NaN
Kitchen	NaN	NaN	NaN	NaN	NaN	NaN
TotRmsAbvGrd	NaN	NaN	NaN	NaN	NaN	0.533723
Fireplaces	NaN	NaN	NaN	NaN	NaN	NaN
GarageYrBlt	NaN	NaN	NaN	NaN	NaN	NaN
GarageCars	NaN	NaN	NaN	NaN	NaN	0.640409
GarageArea	NaN	NaN	NaN	NaN	NaN	0.623431
WoodDeckSF	NaN	NaN	NaN	NaN	NaN	NaN
OpenPorchSF	NaN	NaN	NaN	NaN	NaN	NaN
EnclosedPorch	NaN	NaN	NaN	NaN	NaN	NaN
3SsnPorch	NaN	NaN	NaN	NaN	NaN	NaN
ScreenPorch	1.0	NaN	NaN	NaN	NaN	NaN
PoolArea	NaN	1.0	NaN	NaN	NaN	NaN
MiscVal	NaN	NaN	1.0	NaN	NaN	NaN
MoSold	NaN	NaN	NaN	1.0	NaN	NaN

```
YrSold          NaN      NaN      NaN      NaN      1.0      NaN  
SalePrice       NaN      NaN      NaN      NaN      NaN      1.000000
```

[38 rows x 38 columns]

```
[9]: numerical_cos = dataframe.select_dtypes(include=[np.number])  
sns.pairplot(numerical_cos, diag_kind='kde')  
print(plt.show())
```

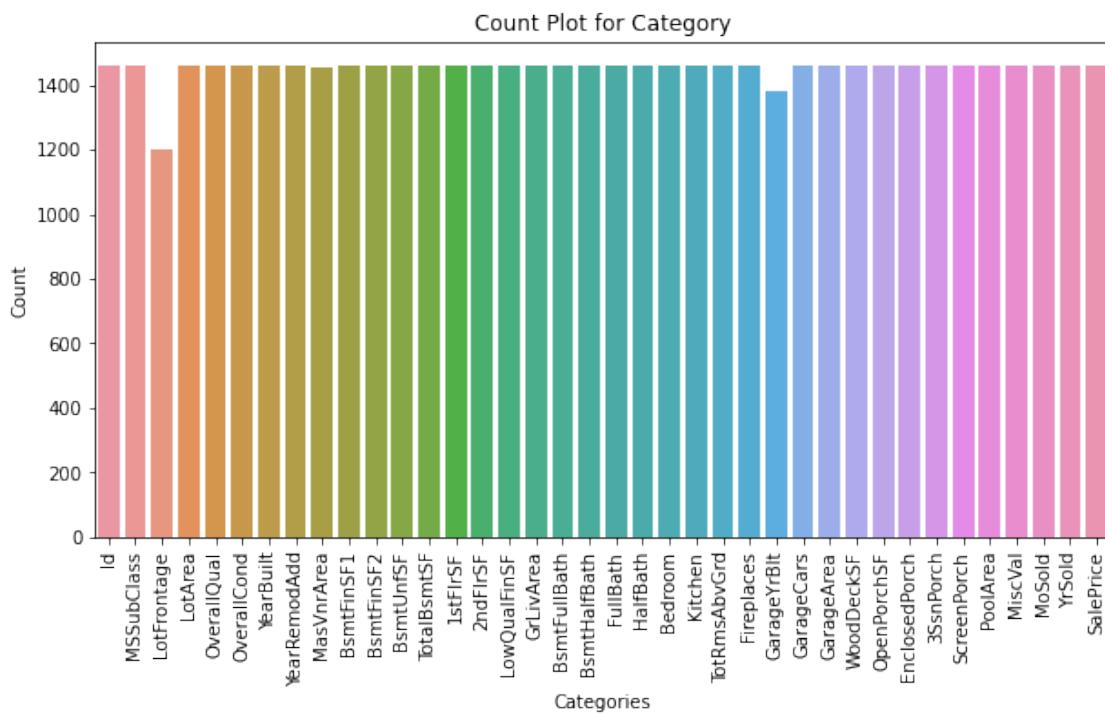


None

```
[18]: missing_values = dataframe.isnull().sum()  
categorical_distribution = dataframe.value_counts()
```

```
[38]: plt.figure(figsize=(10, 5))
sns.countplot(data=dataframe)
plt.title('Count Plot for Category')

plt.xlabel('Categories')
plt.ylabel('Count')
plt.xticks(rotation=90)
plt.show()
```



```
[ ]:
```