

```
In [1]: import pandas as pd
```

```
In [2]: import numpy as np
```

```
In [3]: import matplotlib.pyplot as plt
```

```
In [4]: from matplotlib import style
```

```
In [5]: import seaborn as sns
```

```
In [6]: %matplotlib inline
```

```
In [7]: # Import PEP1.csv file
pep1 = pd.read_csv('C:/Users/User/Desktop/Data Science Notes/Project/PS DS - Appl
```

```
In [8]: pep1.head()
```

```
Out[8]:
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPu
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPu
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPu
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPu
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPu

5 rows × 81 columns

```
In [9]: ## Understand the dataset
```

```
In [10]: # Identify the shape of the dataset
pep1.shape
```

```
Out[10]: (1460, 81)
```

```
In [11]: # Identify variables with null values
         pep1.isna().any()
```

```
Out[11]: Id                False
         MSSubClass         False
         MSZoning           False
         LotFrontage        True
         LotArea            False
         ...
         MoSold            False
         YrSold            False
         SaleType          False
         SaleCondition      False
         SalePrice          False
         Length: 81, dtype: bool
```

```
In [12]: # Identify null values in null specific columns only
         pep1.isna().any()[pep1.isna().any()==True]
```

```
Out[12]: LotFrontage      True
         Alley            True
         MasVnrType       True
         MasVnrArea       True
         BsmtQual         True
         BsmtCond         True
         BsmtExposure     True
         BsmtFinType1     True
         BsmtFinType2     True
         Electrical      True
         FireplaceQu      True
         GarageType       True
         GarageYrBlt      True
         GarageFinish     True
         GarageQual       True
         GarageCond       True
         PoolQC           True
         Fence            True
         MiscFeature      True
         dtype: bool
```

```
In [13]: # Identify count of null values
         pep1.isna().sum()
```

```
Out[13]: Id                0
         MSSubClass         0
         MSZoning           0
         LotFrontage        259
         LotArea            0
         ...
         MoSold            0
         YrSold            0
         SaleType          0
         SaleCondition      0
         SalePrice          0
         Length: 81, dtype: int64
```

```
In [14]: # Identify count of null values in null specific columns only
pep1.isna().sum()[pep1.isna().sum()!=0]
```

```
Out[14]: LotFrontage      259
Alley          1369
MasVnrType      8
MasVnrArea      8
BsmtQual       37
BsmtCond       37
BsmtExposure    38
BsmtFinType1    37
BsmtFinType2    38
Electrical      1
FireplaceQu     690
GarageType      81
GarageYrBlt     81
GarageFinish    81
GarageQual      81
GarageCond      81
PoolQC         1453
Fence          1179
MiscFeature     1406
dtype: int64
```

```
In [15]: # Identify variables with unique values
pep1.nunique()
```

```
Out[15]: Id              1460
MSSubClass              15
MSZoning                5
LotFrontage            110
LotArea               1073
...
MoSold                 12
YrSold                 5
SaleType               9
SaleCondition          6
SalePrice             663
Length: 81, dtype: int64
```

```
In [16]: # Identify variables with unique values in unique specific columns
pep1.nunique()[pep1.nunique()!=0]
```

```
Out[16]: Id                1460
MSSubClass                15
MSZoning                   5
LotFrontage               110
LotArea                   1073
...
MoSold                    12
YrSold                     5
SaleType                   9
SaleCondition              6
SalePrice                 663
Length: 81, dtype: int64
```

```
In [17]: pep1.columns
```

```
Out[17]: Index(['Id', 'MSSubClass', 'MSZoning', 'LotFrontage', 'LotArea', 'Street',
               'Alley', 'LotShape', 'LandContour', 'Utilities', 'LotConfig',
               'LandSlope', 'Neighborhood', 'Condition1', 'Condition2', 'BldgType',
               'HouseStyle', 'OverallQual', 'OverallCond', 'YearBuilt', 'YearRemodAdd',
               'RoofStyle', 'RoofMatl', 'Exterior1st', 'Exterior2nd', 'MasVnrType',
               'MasVnrArea', 'ExterQual', 'ExterCond', 'Foundation', 'BsmtQual',
               'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinSF1',
               'BsmtFinType2', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', 'Heating',
               'HeatingQC', 'CentralAir', 'Electrical', '1stFlrSF', '2ndFlrSF',
               'LowQualFinSF', 'GrLivArea', 'BsmtFullBath', 'BsmtHalfBath', 'FullBath',
               'HalfBath', 'BedroomAbvGr', 'KitchenAbvGr', 'KitchenQual', 'TotRmsAbvGrd',
               'Function1', 'Fireplaces', 'FireplaceQu', 'GarageType', 'GarageYrBlt',
               'GarageFinish', 'GarageCars', 'GarageArea', 'GarageQual', 'GarageCond',
               'PavedDrive', 'WoodDeckSF', 'OpenPorchSF', 'EnclosedPorch', '3SsnPorch',
               'ScreenPorch', 'PoolArea', 'PoolQC', 'Fence', 'MiscFeature', 'MiscVal',
               'MoSold', 'YrSold', 'SaleType', 'SaleCondition', 'SalePrice'],
              dtype='object')
```

```
In [18]: pep1.index
```

```
Out[18]: RangeIndex(start=0, stop=1460, step=1)
```

```
In [19]: ## Generate a separate dataset for numerical and categorical variables
```

```
In [20]: # Dataset for numerical values
```

```
In [21]: num = pep1.select_dtypes(exclude='object')
```

In [22]: num.head()

Out[22]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	YearRemodAdd	Ma
0	1	60	65.0	8450	7	5	2003	2003	
1	2	20	80.0	9600	6	8	1976	1976	
2	3	60	68.0	11250	7	5	2001	2002	
3	4	70	60.0	9550	7	5	1915	1970	
4	5	60	84.0	14260	8	5	2000	2000	

5 rows × 38 columns

In [23]: *# Dataset for categorical values*
catval = pep1.select_dtypes(include='object')

In [24]: catval.head()

Out[24]:

	MSZoning	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood
0	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	CollgCr
1	RL	Pave	NaN	Reg	Lvl	AllPub	FR2	Gtl	Veenker
2	RL	Pave	NaN	IR1	Lvl	AllPub	Inside	Gtl	CollgCr
3	RL	Pave	NaN	IR1	Lvl	AllPub	Corner	Gtl	Crawfor
4	RL	Pave	NaN	IR1	Lvl	AllPub	FR2	Gtl	NoRidge

5 rows × 43 columns

In [25]: *## EDA of numerical variables*

In [26]: *## Missing value treatment*

```
In [27]: # Find Missing values in Numerical Data  
num.isna().any()
```

```
Out[27]: Id                False  
MSSubClass                False  
LotFrontage               True  
LotArea                   False  
OverallQual               False  
OverallCond               False  
YearBuilt                 False  
YearRemodAdd              False  
MasVnrArea                True  
BsmtFinSF1                False  
BsmtFinSF2                False  
BsmtUnfSF                 False  
TotalBsmtSF               False  
1stFlrSF                  False  
2ndFlrSF                  False  
LowQualFinSF              False  
GrLivArea                 False  
BsmtFullBath              False  
BsmtHalfBath              False  
FullBath                  False  
HalfBath                  False  
BedroomAbvGr              False  
KitchenbvGr               False  
TotRmsAbvGrd              False  
Fireplaces                False  
GarageYrBlt               True  
GarageCars                False  
GarageArea                 False  
WoodDeckSF                False  
OpenPorchSF               False  
EnclosedPorch              False  
3SsnPorch                 False  
ScreenPorch                False  
PoolArea                  False  
MiscVal                   False  
MoSold                    False  
YrSold                    False  
SalePrice                 False  
dtype: bool
```

```
In [28]: # Find Missing values columns in Numerical Data having null vales only  
num.isna().any()[num.isna().any()==True]
```

```
Out[28]: LotFrontage      True  
MasVnrArea                True  
GarageYrBlt               True  
dtype: bool
```

```
In [29]: # Find count of Missing values columns in Numerical Data
num.isna().sum()
```

```
Out[29]: Id                0
         MSSubClass        0
         LotFrontage      259
         LotArea           0
         OverallQual       0
         OverallCond       0
         YearBuilt         0
         YearRemodAdd      0
         MasVnrArea        8
         BsmtFinSF1        0
         BsmtFinSF2        0
         BsmtUnfSF         0
         TotalBsmtSF       0
         1stFlrSF          0
         2ndFlrSF          0
         LowQualFinSF      0
         GrLivArea         0
         BsmtFullBath       0
         BsmtHalfBath       0
         FullBath           0
         HalfBath           0
         BedroomAbvGr       0
         KitchenAbvGr       0
         TotRmsAbvGrd       0
         Fireplaces         0
         GarageYrBlt       81
         GarageCars         0
         GarageArea         0
         WoodDeckSF         0
         OpenPorchSF        0
         EnclosedPorch      0
         3SsnPorch          0
         ScreenPorch        0
         PoolArea           0
         MiscVal            0
         MoSold             0
         YrSold             0
         SalePrice          0
         dtype: int64
```

```
In [30]: # Find count of Missing values columns in Numerical Data having null vales only
num.isna().sum()[num.isna().sum()!=0]
```

```
Out[30]: LotFrontage      259
         MasVnrArea        8
         GarageYrBlt      81
         dtype: int64
```

```
In [31]: num['LotFrontage']
```

```
Out[31]: 0      65.0  
         1      80.0  
         2      68.0  
         3      60.0  
         4      84.0  
         ...  
        1455     62.0  
        1456     85.0  
        1457     66.0  
        1458     68.0  
        1459     75.0  
        Name: LotFrontage, Length: 1460, dtype: float64
```

```
In [32]: # Remove missing values from columnns having null values  
num.dropna(subset=['LotFrontage', 'MasVnrArea', 'GarageYrBlt'], inplace=True)
```

```
In [33]: num.isna().sum()[num.isna().sum()!=0]
```

```
Out[33]: Series([], dtype: int64)
```

```
In [34]: # Identify the skewness and distribution
```



```
In [35]: num.skew(axis=0,skipna=True)
```

```
Out[35]: Id                0.018663
MSSubClass          1.412907
LotFrontage         2.251197
LotArea            15.608113
OverallQual         0.287800
OverallCond         0.846451
YearBuilt          -0.618350
YearRemodAdd       -0.565757
MasVnrArea          2.706945
BsmtFinSF1          1.934077
BsmtFinSF2          4.399358
BsmtUnfSF           0.875774
TotalBsmtSF         1.754916
1stFlrSF            1.363783
2ndFlrSF            0.807411
LowQualFinSF        10.020823
GrLivArea           1.549961
BsmtFullBath         0.568804
BsmtHalfBath         4.107874
FullBath             0.015822
HalfBath             0.638178
BedroomAbvGr         0.074427
KitchenbvGr          4.822542
TotRmsAbvGrd         0.723117
Fireplaces           0.643698
GarageYrBlt        -0.641738
GarageCars           0.206017
GarageArea           0.733894
WoodDeckSF           1.549793
OpenPorchSF          2.403928
EnclosedPorch        3.173250
3SsnPorch           10.854868
ScreenPorch          4.019111
PoolArea            13.783823
MiscVal              9.699989
MoSold               0.173039
YrSold               0.106730
SalePrice            1.933615
dtype: float64
```

```
In [36]: num['LotArea']
```

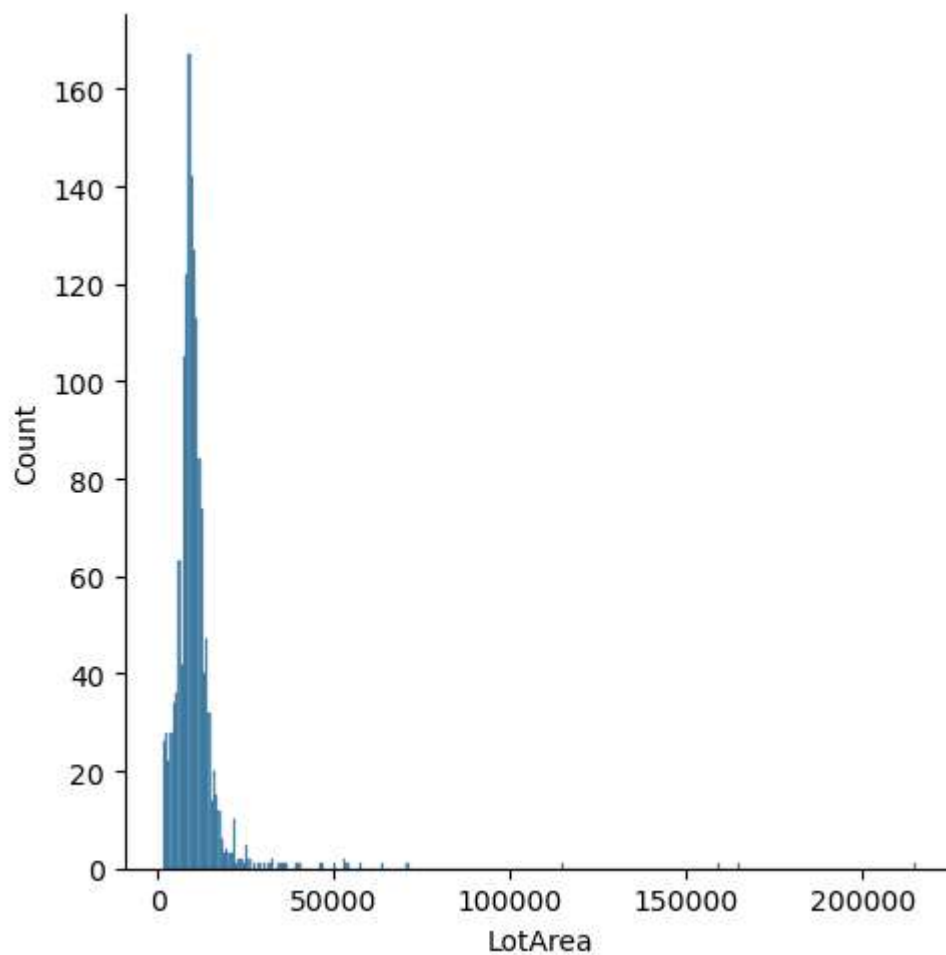
```
Out[36]: 0          8450
1          9600
2         11250
3          9550
4         14260
...
1455        7917
1456       13175
1457        9042
1458        9717
1459        9937
Name: LotArea, Length: 1121, dtype: int64
```

```
In [37]: num['LotArea'].max()
```

```
Out[37]: 215245
```

```
In [38]: # Plot Distribution of Numerical Variables  
sns.displot(pep1.LotArea)
```

```
Out[38]: <seaborn.axisgrid.FacetGrid at 0xef2fd3a3a0>
```



```
In [39]: # Identify significant variables using a correlation matrix
```

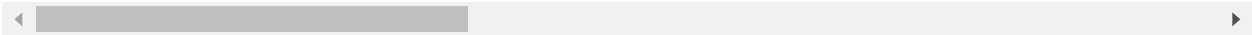
In [40]: num.corr()

Out[40]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBui
Id	1.000000	0.021937	-0.013289	-0.040711	-0.058269	0.004387	-0.02086
MSSubClass	0.021937	1.000000	-0.386940	-0.198096	0.029522	-0.087859	0.02580
LotFrontage	-0.013289	-0.386940	1.000000	0.421184	0.241322	-0.046312	0.10972
LotArea	-0.040711	-0.198096	0.421184	1.000000	0.167525	-0.034348	0.02920
OverallQual	-0.058269	0.029522	0.241322	0.167525	1.000000	-0.163157	0.58938
OverallCond	0.004387	-0.087859	-0.046312	-0.034348	-0.163157	1.000000	-0.42646
YearBuilt	-0.020862	0.025800	0.109726	0.029205	0.589385	-0.426462	1.00000
YearRemodAdd	-0.027664	0.006645	0.086414	0.026848	0.570757	0.039402	0.62317
MasVnrArea	-0.073472	0.040240	0.189969	0.106115	0.423988	-0.166762	0.33219
BsmtFinSF1	-0.013751	-0.070389	0.241352	0.230441	0.249500	-0.054788	0.23694
BsmtFinSF2	0.012544	-0.075439	0.049305	0.138234	-0.068506	0.042314	-0.05441
BsmtUnfSF	-0.012985	-0.145582	0.115306	0.011288	0.322663	-0.148630	0.17754
TotalBsmtSF	-0.023129	-0.247781	0.387620	0.302554	0.563960	-0.192762	0.40913
1stFlrSF	-0.008046	-0.252249	0.451085	0.329679	0.514453	-0.164251	0.30887
2ndFlrSF	-0.002346	0.319328	0.075004	0.074612	0.273197	0.005985	-0.01162
LowQualFinSF	-0.039933	0.024704	0.011148	0.020039	-0.008118	0.048720	-0.16435
GrLivArea	-0.011068	0.083365	0.396306	0.307164	0.607466	-0.112231	0.20496
BsmtFullBath	0.026113	-0.014681	0.118088	0.179052	0.126834	-0.060943	0.18280
BsmtHalfBath	-0.026774	0.012310	0.000434	-0.014282	-0.053283	0.122960	-0.04964
FullBath	0.007220	0.131278	0.185785	0.129073	0.576875	-0.229848	0.50049
HalfBath	-0.010409	0.203971	0.045678	0.045183	0.251690	-0.079023	0.22000
BedroomAbvGr	0.039831	-0.032971	0.270404	0.137269	0.094882	0.004643	-0.06158
KitchenAbvGr	0.025913	0.266012	-0.003546	-0.018942	-0.178735	-0.092644	-0.17192
TotRmsAbvGrd	0.020012	0.047209	0.348421	0.237918	0.451008	-0.096901	0.12141
Fireplaces	-0.018273	-0.031122	0.260321	0.255755	0.415294	-0.022290	0.13307
GarageYrBlt	-0.002039	0.054701	0.069878	0.013731	0.560425	-0.343206	0.82352
GarageCars	-0.008125	-0.027411	0.286587	0.172428	0.593803	-0.267859	0.53256
GarageArea	-0.025889	-0.092607	0.356851	0.211362	0.550659	-0.226347	0.47128
WoodDeckSF	-0.025060	-0.017988	0.082166	0.133576	0.282512	-0.010835	0.23854
OpenPorchSF	-0.001972	0.004054	0.161815	0.099170	0.340679	-0.076273	0.23543
EnclosedPorch	0.009935	-0.017790	0.014261	-0.023631	-0.144344	0.062748	-0.39269
3SsnPorch	-0.066833	-0.039739	0.069716	0.012520	0.017331	-0.006861	0.02794
ScreenPorch	0.015183	-0.021789	0.035906	0.072517	0.055296	0.087030	-0.06369
PoolArea	0.048010	0.003166	0.211746	0.109147	0.080131	-0.023566	0.00671
MiscVal	0.045799	-0.040689	0.001471	0.012790	-0.062064	0.119772	-0.09697

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBui
MoSold	-0.000570	-0.027170	0.018815	0.008998	0.079895	-0.014236	0.01378
YrSold	0.013407	-0.012448	0.013267	-0.006904	-0.008903	0.041003	-0.00458
SalePrice	-0.047122	-0.088032	0.344270	0.299962	0.797881	-0.124391	0.52539

38 rows × 38 columns



In []:

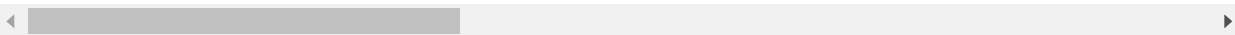
```
In [41]: # Column Specific Correlation
num.corr(method='pearson')
```

Out[41]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBui
Id	1.000000	0.021937	-0.013289	-0.040711	-0.058269	0.004387	-0.02086
MSSubClass	0.021937	1.000000	-0.386940	-0.198096	0.029522	-0.087859	0.02580
LotFrontage	-0.013289	-0.386940	1.000000	0.421184	0.241322	-0.046312	0.10972
LotArea	-0.040711	-0.198096	0.421184	1.000000	0.167525	-0.034348	0.02920
OverallQual	-0.058269	0.029522	0.241322	0.167525	1.000000	-0.163157	0.58938
OverallCond	0.004387	-0.087859	-0.046312	-0.034348	-0.163157	1.000000	-0.42646
YearBuilt	-0.020862	0.025800	0.109726	0.029205	0.589385	-0.426462	1.00000
YearRemodAdd	-0.027664	0.006645	0.086414	0.026848	0.570757	0.039402	0.62317
MasVnrArea	-0.073472	0.040240	0.189969	0.106115	0.423988	-0.166762	0.33219
BsmtFinSF1	-0.013751	-0.070389	0.241352	0.230441	0.249500	-0.054788	0.23694
BsmtFinSF2	0.012544	-0.075439	0.049305	0.138234	-0.068506	0.042314	-0.05441
BsmtUnfSF	-0.012985	-0.145582	0.115306	0.011288	0.322663	-0.148630	0.17754
TotalBsmtSF	-0.023129	-0.247781	0.387620	0.302554	0.563960	-0.192762	0.40913
1stFlrSF	-0.008046	-0.252249	0.451085	0.329679	0.514453	-0.164251	0.30887
2ndFlrSF	-0.002346	0.319328	0.075004	0.074612	0.273197	0.005985	-0.01162
LowQualFinSF	-0.039933	0.024704	0.011148	0.020039	-0.008118	0.048720	-0.16435
GrLivArea	-0.011068	0.083365	0.396306	0.307164	0.607466	-0.112231	0.20496
BsmtFullBath	0.026113	-0.014681	0.118088	0.179052	0.126834	-0.060943	0.18280
BsmtHalfBath	-0.026774	0.012310	0.000434	-0.014282	-0.053283	0.122960	-0.04964
FullBath	0.007220	0.131278	0.185785	0.129073	0.576875	-0.229848	0.50049
HalfBath	-0.010409	0.203971	0.045678	0.045183	0.251690	-0.079023	0.22000
BedroomAbvGr	0.039831	-0.032971	0.270404	0.137269	0.094882	0.004643	-0.06158
KitchenAbvGr	0.025913	0.266012	-0.003546	-0.018942	-0.178735	-0.092644	-0.17192
TotRmsAbvGrd	0.020012	0.047209	0.348421	0.237918	0.451008	-0.096901	0.12141
Fireplaces	-0.018273	-0.031122	0.260321	0.255755	0.415294	-0.022290	0.13307
GarageYrBlt	-0.002039	0.054701	0.069878	0.013731	0.560425	-0.343206	0.82352
GarageCars	-0.008125	-0.027411	0.286587	0.172428	0.593803	-0.267859	0.53256
GarageArea	-0.025889	-0.092607	0.356851	0.211362	0.550659	-0.226347	0.47128
WoodDeckSF	-0.025060	-0.017988	0.082166	0.133576	0.282512	-0.010835	0.23854
OpenPorchSF	-0.001972	0.004054	0.161815	0.099170	0.340679	-0.076273	0.23543
EnclosedPorch	0.009935	-0.017790	0.014261	-0.023631	-0.144344	0.062748	-0.39269
3SsnPorch	-0.066833	-0.039739	0.069716	0.012520	0.017331	-0.006861	0.02794
ScreenPorch	0.015183	-0.021789	0.035906	0.072517	0.055296	0.087030	-0.06369
PoolArea	0.048010	0.003166	0.211746	0.109147	0.080131	-0.023566	0.00671

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt
MiscVal	0.045799	-0.040689	0.001471	0.012790	-0.062064	0.119772	-0.09697
MoSold	-0.000570	-0.027170	0.018815	0.008998	0.079895	-0.014236	0.01378
YrSold	0.013407	-0.012448	0.013267	-0.006904	-0.008903	0.041003	-0.00458
SalePrice	-0.047122	-0.088032	0.344270	0.299962	0.797881	-0.124391	0.52539

38 rows × 38 columns



```
In [42]: # Specific Variable Correlation
var = num.corr().SalePrice[(num.corr().SalePrice>0.5)|(num.corr().SalePrice<0.5)]
```

```
In [43]: var
```

```
Out[43]: Index(['Id', 'MSSubClass', 'LotFrontage', 'LotArea', 'OverallQual',
               'OverallCond', 'YearBuilt', 'YearRemodAdd', 'MasVnrArea', 'BsmtFinSF1',
               'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', '1stFlrSF', '2ndFlrSF',
               'LowQualFinSF', 'GrLivArea', 'BsmtFullBath', 'BsmtHalfBath', 'FullBath',
               'HalfBath', 'BedroomAbvGr', 'KitchenAbvGr', 'TotRmsAbvGrd', 'Fireplaces',
               'GarageYrBlt', 'GarageCars', 'GarageArea', 'WoodDeckSF', 'OpenPorchSF',
               'EnclosedPorch', '3SsnPorch', 'ScreenPorch', 'PoolArea', 'MiscVal',
               'MoSold', 'YrSold', 'SalePrice'],
              dtype='object')
```

```
In [44]: pep1[var].nunique()
```

```
Out[44]: Id                1460
      MSSubClass           15
      LotFrontage          110
      LotArea              1073
      OverallQual           10
      OverallCond           9
      YearBuilt             112
      YearRemodAdd           61
      MasVnrArea            327
      BsmtFinSF1             637
      BsmtFinSF2            144
      BsmtUnfSF              780
      TotalBsmtSF           721
      1stFlrSF              753
      2ndFlrSF              417
      LowQualFinSF           24
      GrLivArea             861
      BsmtFullBath           4
      BsmtHalfBath           3
      FullBath               4
      HalfBath               3
      BedroomAbvGr           8
      KitchenbvGr            4
      TotRmsAbvGrd           12
      Fireplaces             4
      GarageYrBlt           97
      GarageCars             5
      GarageArea            441
      WoodDeckSF            274
      OpenPorchSF           202
      EnclosedPorch          120
      3SsnPorch              20
      ScreenPorch            76
      PoolArea               8
      MiscVal                21
      MoSold                 12
      YrSold                  5
      SalePrice              663
      dtype: int64
```

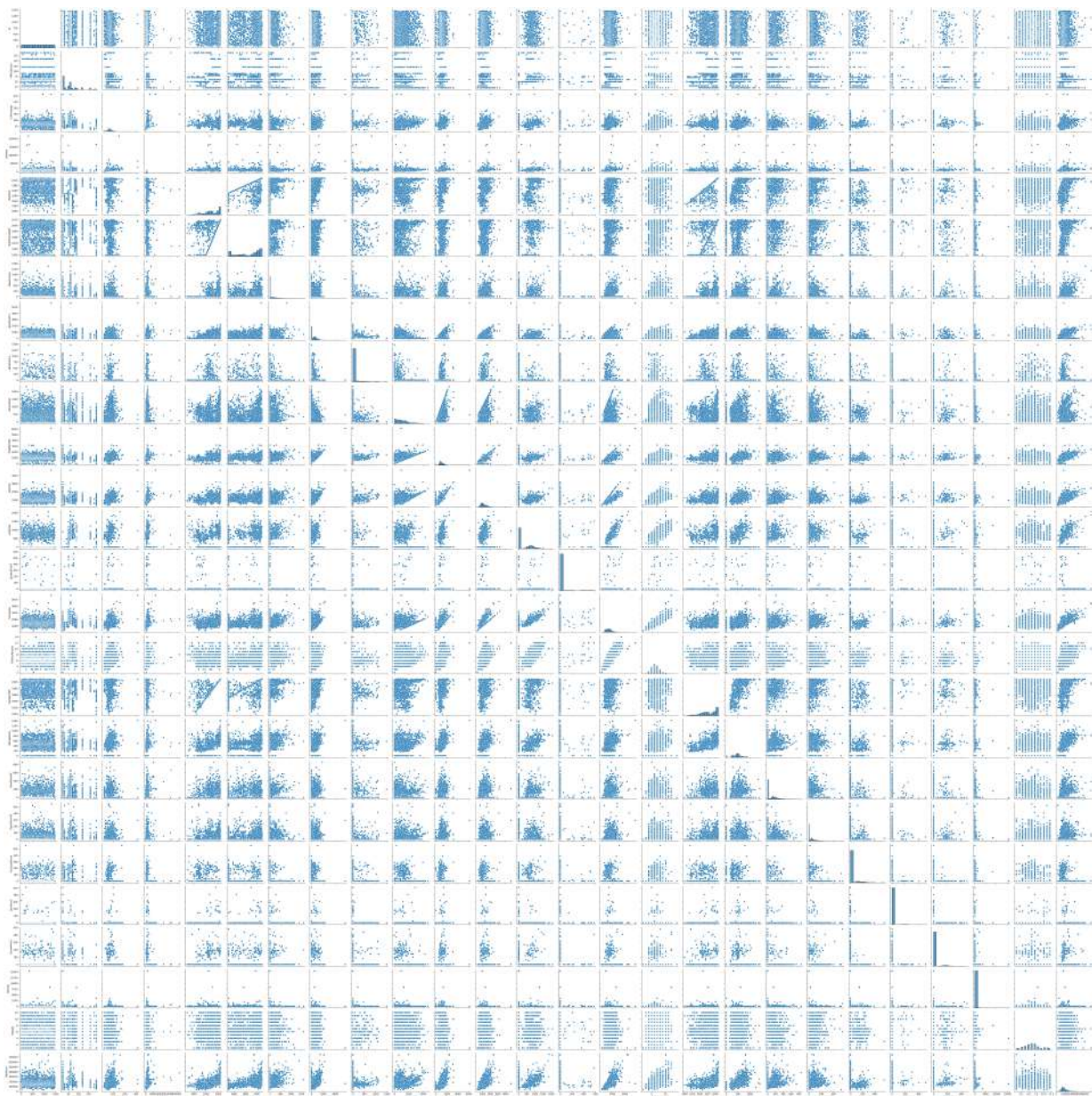
Type *Markdown* and LaTeX: α^2

```
In [45]: var2 = pep1[var].nunique()[pep1[var].nunique()>10].index
```



```
In [47]: sns.pairplot(peg1[var2])
```

```
Out[47]: <seaborn.axisgrid.PairGrid at 0xe9cbca5fa0>
```



```
In [46]: ## EDA of categorical variables
```

```
In [47]: # Missing value treatment
```


In [48]:

catval

Out[48]:

	MSZoning	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood
0	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	Coll
1	RL	Pave	NaN	Reg	Lvl	AllPub	FR2	Gtl	Veer
2	RL	Pave	NaN	IR1	Lvl	AllPub	Inside	Gtl	Coll
3	RL	Pave	NaN	IR1	Lvl	AllPub	Corner	Gtl	Crav
4	RL	Pave	NaN	IR1	Lvl	AllPub	FR2	Gtl	NoRi
...	
1455	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	Gill
1456	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	NWAr
1457	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	Crav
1458	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	r
1459	RL	Pave	NaN	Reg	Lvl	AllPub	Inside	Gtl	Edwa

1460 rows × 43 columns

```
In [49]: catval.isna().any()
```

```
Out[49]: MSZoning      False
Street      False
Alley       True
LotShape    False
LandContour False
Utilities   False
LotConfig   False
LandSlope   False
Neighborhood False
Condition1  False
Condition2  False
BldgType    False
HouseStyle  False
RoofStyle   False
RoofMatl    False
Exterior1st False
Exterior2nd False
MasVnrType  True
ExterQual   False
ExterCond   False
Foundation  False
BsmtQual    True
BsmtCond    True
BsmtExposure True
BsmtFinType1 True
BsmtFinType2 True
Heating     False
HeatingQC   False
CentralAir  False
Electrical  True
KitchenQual False
Function    False
FireplaceQu True
GarageType  True
GarageFinish True
GarageQual  True
GarageCond  True
PavedDrive  False
PoolQC      True
Fence       True
MiscFeature  True
SaleType     False
SaleCondition False
dtype: bool
```

```
In [50]: catval.isna().any()[catval.isna().any()==True]
```

```
Out[50]: Alley                True
MasVnrType                True
BsmtQual                  True
BsmtCond                  True
BsmtExposure              True
BsmtFinType1              True
BsmtFinType2              True
Electrical                True
FireplaceQu               True
GarageType                 True
GarageFinish              True
GarageQual                True
GarageCond                True
PoolQC                    True
Fence                     True
MiscFeature               True
dtype: bool
```

```
In [51]: catval.isna().sum()[catval.isna().sum()!=0]
```

```
Out[51]: Alley                1369
MasVnrType                   8
BsmtQual                    37
BsmtCond                     37
BsmtExposure                 38
BsmtFinType1                 37
BsmtFinType2                 38
Electrical                   1
FireplaceQu                  690
GarageType                   81
GarageFinish                 81
GarageQual                   81
GarageCond                   81
PoolQC                       1453
Fence                        1179
MiscFeature                  1406
dtype: int64
```

```
In [ ]:
```

Type *Markdown* and LaTeX: α^2

```
In [53]: # Count plot and box plot for bivariate analysis
```

```
In [52]: round(catval.isna().sum()[catval.isna().sum()!=0]/1500*100,2)
```

```
Out[52]: Alley                91.27
MasVnrType                   0.53
BsmtQual                     2.47
BsmtCond                     2.47
BsmtExposure                 2.53
BsmtFinType1                 2.47
BsmtFinType2                 2.53
Electrical                   0.07
FireplaceQu                 46.00
GarageType                   5.40
GarageFinish                 5.40
GarageQual                   5.40
GarageCond                   5.40
PoolQC                      96.87
Fence                       78.60
MiscFeature                  93.73
dtype: float64
```

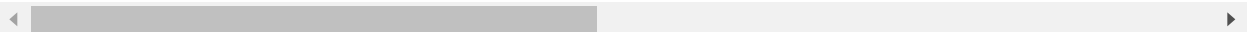
```
In [53]: cat1 = catval.isna().sum()[catval.isna().sum()!=0]/1500*100,2
```

```
In [54]: pep1.head()
```

```
Out[54]:
```

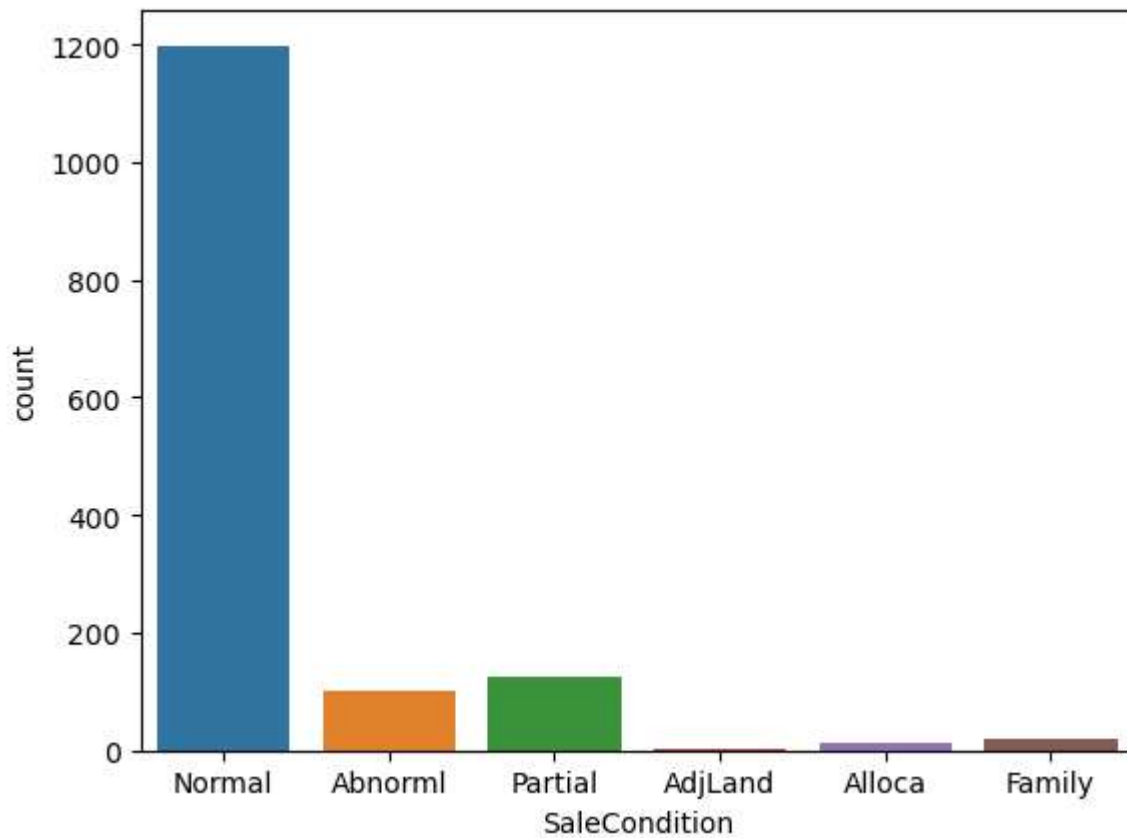
	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPu
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPu
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPu
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPu
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPu

5 rows × 81 columns



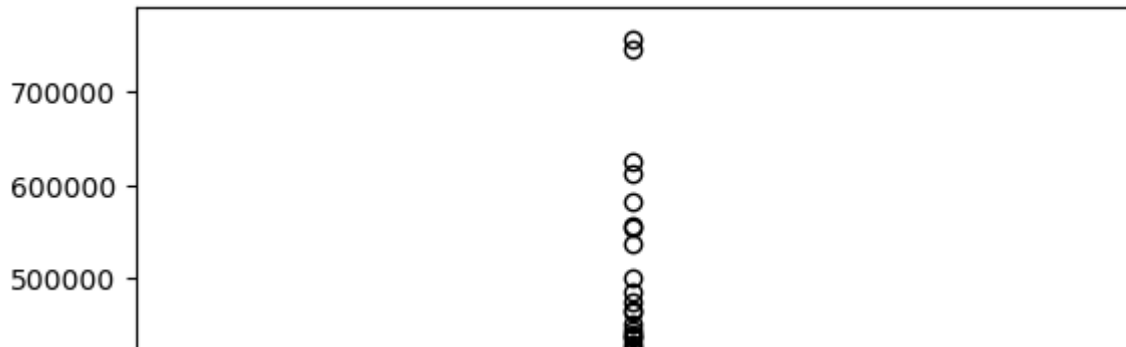
```
In [56]: # Count Plot  
sns.countplot(x=catsal['SaleCondition'])
```

```
Out[56]: <AxesSubplot:xlabel='SaleCondition', ylabel='count'>
```



```
In [57]: # Box Plot
plt.boxplot(pep1.SalePrice)
```

```
Out[57]: {'whiskers': [<matplotlib.lines.Line2D at 0xef2f9841f0>,
<matplotlib.lines.Line2D at 0xef2f9844c0>],
'caps': [<matplotlib.lines.Line2D at 0xef2f984790>,
<matplotlib.lines.Line2D at 0xef2f984a60>],
'boxes': [<matplotlib.lines.Line2D at 0xef2f975ee0>],
'medians': [<matplotlib.lines.Line2D at 0xef2f984d30>],
'fliers': [<matplotlib.lines.Line2D at 0xef2f993040>],
'means': []}
```

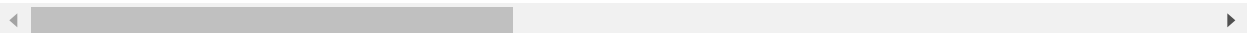


```
In [58]: num.head()
```

```
Out[58]:
```

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	YearRemodAdd	Ma
0	1	60	65.0	8450	7	5	2003	2003	
1	2	20	80.0	9600	6	8	1976	1976	
2	3	60	68.0	11250	7	5	2001	2002	
3	4	70	60.0	9550	7	5	1915	1970	
4	5	60	84.0	14260	8	5	2000	2000	

5 rows × 38 columns



catval.head()

```
In [ ]:
```

```
In [ ]:
```

Type *Markdown* and LaTeX: α^2

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```

In []:

In []:

In []:

In []:

In []:

In []: