In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
import io
%cd "C:\Users\gouth\OneDrive\Desktop\Big Mart\HousePrice"
```

```
C:\Users\gouth\OneDrive\Desktop\Big Mart\HousePrice
```

In [3]:

```
housetrain=pd.read_csv("train.csv")
```

In [4]:

```
housetest=pd.read_csv("test.csv")
```

In [5]:

```
print(housetrain.shape)
print(housetest.shape)
```

```
(1460, 81)
(1459, 80)
```

In [6]:

```
housetrain.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1460 entries, 0 to 1459
Data columns (total 81 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Id             1460 non-null   int64  
 1   MSSubClass     1460 non-null   int64  
 2   MSZoning       1460 non-null   object 
 3   LotFrontage    1201 non-null   float64
 4   LotArea        1460 non-null   int64  
 5   Street         1460 non-null   object 
 6   Alley          91 non-null     object 
 7   LotShape       1460 non-null   object 
 8   LandContour    1460 non-null   object 
 9   Utilities      1460 non-null   object 
 10  LotConfig      1460 non-null   object 
 11  LandSlope      1460 non-null   object 
 12  Neighborhood   1460 non-null   object 
 13  Condition1     1460 non-null   object 
 14  Condition2     1460 non-null   object 
 15  BldgType       1460 non-null   object 
 16  HouseStyle     1460 non-null   object 
 17  OverallQual    1460 non-null   int64  
 18  OverallCond    1460 non-null   int64  
 19  YearBuilt      1460 non-null   int64  
 20  YearRemodAdd   1460 non-null   int64  
 21  RoofStyle      1460 non-null   object 
 22  RoofMatl       1460 non-null   object 
 23  Exterior1st    1460 non-null   object 
 24  Exterior2nd    1460 non-null   object 
 25  MasVnrType     1452 non-null   object 
 26  MasVnrArea     1452 non-null   float64
 27  ExterQual      1460 non-null   object 
 28  ExterCond      1460 non-null   object 
 29  Foundation     1460 non-null   object 
 30  BsmtQual       1423 non-null   object 
 31  BsmtCond       1423 non-null   object 
 32  BsmtExposure   1422 non-null   object 
 33  BsmtFinType1   1423 non-null   object 
 34  BsmtFinSF1     1460 non-null   int64  
 35  BsmtFinType2   1422 non-null   object 
 36  BsmtFinSF2     1460 non-null   int64  
 37  BsmtUnfSF      1460 non-null   int64  
 38  TotalBsmtSF    1460 non-null   int64  
 39  Heating        1460 non-null   object 
 40  HeatingQC      1460 non-null   object 
 41  CentralAir     1460 non-null   object 
 42  Electrical     1459 non-null   object 
 43  1stFlrSF       1460 non-null   int64  
 44  2ndFlrSF       1460 non-null   int64  
 45  LowQualFinSF   1460 non-null   int64  
 46  GrLivArea      1460 non-null   int64  
 47  BsmtFullBath   1460 non-null   int64  
 48  BsmtHalfBath   1460 non-null   int64  
 49  FullBath       1460 non-null   int64  
 50  HalfBath       1460 non-null   int64  
 51  BedroomAbvGr   1460 non-null   int64  
 52  KitchenAbvGr   1460 non-null   int64  
 53  KitchenQual    1460 non-null   object 
 54  TotRmsAbvGrd   1460 non-null   int64  
 55  Functional     1460 non-null   object 
 56  Fireplaces     1460 non-null   int64  
 57  FireplaceQu    770 non-null    object 
 58  GarageType     1379 non-null   object 
 59  GarageYrBlt    1379 non-null   float64
 60  GarageFinish   1379 non-null   object 
 61  GarageCars     1460 non-null   int64  
 62  GarageArea     1460 non-null   int64  
 63  GarageQual     1379 non-null   object 
 64  GarageCond     1379 non-null   object 
 65  PavedDrive     1460 non-null   object 
 66  WoodDeckSF     1460 non-null   int64  
 67  OpenPorchSF    1460 non-null   int64  
 68  EnclosedPorch  1460 non-null   int64  
 69  3SsnPorch      1460 non-null   int64  
 70  ScreenPorch    1460 non-null   int64  
 71  PoolArea       1460 non-null   int64  
 72  PoolQC         7 non-null      object 
 73  Fence          281 non-null    object 
 74  MiscFeature    54 non-null     object 
 75  MiscVal        1460 non-null   int64  
 76  MoSold         1460 non-null   int64  
 77  YrSold         1460 non-null   int64  
 78  SaleType       1460 non-null   object 
 79  SaleCondition  1460 non-null   object 
 80  SalePrice      1460 non-null   int64  
dtypes: float64(3), int64(35), object(43)
memory usage: 924.0+ KB
```

In [7]:

```
housetest.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1459 entries, 0 to 1458
Data columns (total 80 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Id             1459 non-null   int64  
 1   MSSubClass     1459 non-null   int64  
 2   MSZoning       1455 non-null   object 
 3   LotFrontage    1232 non-null   float64
 4   LotArea        1459 non-null   int64  
 5   Street         1459 non-null   object 
 6   Alley          107 non-null    object 
 7   LotShape       1459 non-null   object 
 8   LandContour    1459 non-null   object 
 9   Utilities      1457 non-null   object 
 10  LotConfig      1459 non-null   object 
 11  LandSlope      1459 non-null   object 
 12  Neighborhood   1459 non-null   object 
 13  Condition1     1459 non-null   object 
 14  Condition2     1459 non-null   object 
 15  BldgType       1459 non-null   object 
 16  HouseStyle     1459 non-null   object 
 17  OverallQual    1459 non-null   int64  
 18  OverallCond    1459 non-null   int64  
 19  YearBuilt      1459 non-null   int64  
 20  YearRemodAdd   1459 non-null   int64  
 21  RoofStyle      1459 non-null   object 
 22  RoofMatl       1459 non-null   object 
 23  Exterior1st    1458 non-null   object 
 24  Exterior2nd    1458 non-null   object 
 25  MasVnrType     1443 non-null   object 
 26  MasVnrArea     1444 non-null   float64
 27  ExterQual      1459 non-null   object 
 28  ExterCond      1459 non-null   object 
 29  Foundation     1459 non-null   object 
 30  BsmtQual       1415 non-null   object 
 31  BsmtCond       1414 non-null   object 
 32  BsmtExposure   1415 non-null   object 
 33  BsmtFinType1   1417 non-null   object 
 34  BsmtFinSF1     1458 non-null   float64
 35  BsmtFinType2   1417 non-null   object 
 36  BsmtFinSF2     1458 non-null   float64
 37  BsmtUnfSF      1458 non-null   float64
 38  TotalBsmtSF    1458 non-null   float64
 39  Heating        1459 non-null   object 
 40  HeatingQC      1459 non-null   object 
 41  CentralAir     1459 non-null   object 
 42  Electrical     1459 non-null   object 
 43  1stFlrSF       1459 non-null   int64  
 44  2ndFlrSF       1459 non-null   int64  
 45  LowQualFinSF   1459 non-null   int64  
 46  GrLivArea      1459 non-null   int64  
 47  BsmtFullBath   1457 non-null   float64
 48  BsmtHalfBath   1457 non-null   float64
 49  FullBath       1459 non-null   int64  
 50  HalfBath       1459 non-null   int64  
 51  BedroomAbvGr   1459 non-null   int64  
 52  KitchenAbvGr   1459 non-null   int64  
 53  KitchenQual    1458 non-null   object 
 54  TotRmsAbvGrd   1459 non-null   int64  
 55  Functional     1457 non-null   object 
 56  Fireplaces     1459 non-null   int64  
 57  FireplaceQu    729 non-null    object 
 58  GarageType     1383 non-null   object 
 59  GarageYrBlt    1381 non-null   float64
 60  GarageFinish   1381 non-null   object 
 61  GarageCars     1458 non-null   float64
 62  GarageArea     1458 non-null   float64
 63  GarageQual     1381 non-null   object 
 64  GarageCond     1381 non-null   object 
 65  PavedDrive     1459 non-null   object 
 66  WoodDeckSF     1459 non-null   int64  
 67  OpenPorchSF    1459 non-null   int64  
 68  EnclosedPorch  1459 non-null   int64  
 69  3SsnPorch      1459 non-null   int64  
 70  ScreenPorch    1459 non-null   int64  
 71  PoolArea       1459 non-null   int64  
 72  PoolQC         3 non-null      object 
 73  Fence          290 non-null    object 
 74  MiscFeature    51 non-null     object 
 75  MiscVal        1459 non-null   int64  
 76  MoSold         1459 non-null   int64  
 77  YrSold         1459 non-null   int64  
 78  SaleType       1458 non-null   object 
 79  SaleCondition  1459 non-null   object 
dtypes: float64(11), int64(26), object(43)
memory usage: 912.0+ KB
```

In [8]:

```
housetest["SalePrice"]="test"
```

In [9]:

```
combinedf=pd.concat([housetrain,housetest],axis=0)
```

In [10]:

```
pd.set_option("display.max_rows",82)
```

In [11]:

```
combinedf.isnull().sum().sort_values(ascending=False)
```

Out[11]:

```
PoolQC           2909
MiscFeature      2814
Alley            2721
Fence            2348
FireplaceQu      1420
LotFrontage       486
GarageFinish      159
GarageQual        159
GarageCond        159
GarageYrBlt       159
GarageType        157
BsmtExposure       82
BsmtCond           82
BsmtQual           81
BsmtFinType2       80
BsmtFinType1       79
MasVnrType         24
MasVnrArea         23
MSZoning            4
Functional          2
Utilities           2
BsmtHalfBath        2
BsmtFullBath        2
GarageArea          1
BsmtFinSF1          1
SaleType            1
GarageCars          1
BsmtUnfSF           1
Electrical          1
Exterior2nd         1
Exterior1st         1
KitchenQual         1
TotalBsmtSF         1
BsmtFinSF2          1
TotRmsAbvGrd        0
Fireplaces          0
Id                  0
BedroomAbvGr        0
PavedDrive          0
WoodDeckSF          0
OpenPorchSF         0
EnclosedPorch       0
3SsnPorch           0
ScreenPorch         0
PoolArea            0
MiscVal             0
MoSold              0
YrSold              0
SaleCondition       0
KitchenAbvGr        0
HeatingQC           0
HalfBath            0
FullBath            0
LotArea             0
Street              0
LotShape            0
LandContour         0
LotConfig           0
LandSlope           0
Neighborhood        0
Condition1          0
Condition2          0
BldgType            0
HouseStyle          0
OverallQual         0
OverallCond         0
YearBuilt           0
YearRemodAdd        0
RoofStyle           0
RoofMatl            0
ExterQual           0
ExterCond           0
Foundation          0
Heating             0
MSSubClass          0
CentralAir          0
1stFlrSF            0
2ndFlrSF            0
LowQualFinSF        0
GrLivArea           0
SalePrice           0
dtype: int64
```

In [12]:

```
objectcols=combinedf.select_dtypes(include=["object"])
numericcols=combinedf.select_dtypes(include=np.number)
```

In [13]:

```
print(objectcols.shape)
print(numericcols.shape)
```

```
(2919, 44)
(2919, 37)
```

In [14]:

```
objectcols.isnull().sum().sort_values(ascending=False)
```

Out[14]:

```
PoolQC           2909
MiscFeature      2814
Alley            2721
Fence            2348
FireplaceQu      1420
GarageCond        159
GarageQual        159
GarageFinish      159
GarageType        157
BsmtCond           82
BsmtExposure       82
BsmtQual           81
BsmtFinType2       80
BsmtFinType1       79
MasVnrType         24
MSZoning            4
Utilities           2
Functional          2
Exterior2nd         1
Electrical          1
SaleType            1
Exterior1st         1
KitchenQual         1
RoofStyle           0
Neighborhood        0
SaleCondition       0
ExterQual           0
LotShape            0
LandContour         0
ExterCond           0
PavedDrive          0
LotConfig           0
LandSlope           0
Condition1          0
HouseStyle          0
Condition2          0
Foundation          0
RoofMatl            0
Street              0
CentralAir          0
HeatingQC           0
Heating             0
BldgType            0
SalePrice           0
dtype: int64
```

In [15]:

```
combinedf.columns
```

Out[15]:

```
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', 'Functional', '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 [16]:

```
notavailablecols=['PoolQC','MiscFeature','Alley','Fence','FireplaceQu']
```

In [17]:

```
for col in notavailablecols:objectcols[col]=objectcols[col].fillna("notavailable")
```

In [18]:

```
garage_list=[col for col in objectcols if col.startswith("Gar")]
garage_list
```

Out[18]:

```
['GarageType', 'GarageFinish', 'GarageQual', 'GarageCond']
```

In [19]:

```
for col in garage_list:
    freq=objectcols[col].value_counts(dropna=False)
    print(freq)
```

```
Attchd     1723
Detchd      779
BuiltIn     186
NaN         157
Basment      36
2Types       23
CarPort      15
Name: GarageType, dtype: int64
Unf    1230
RFn     811
Fin     719
NaN     159
Name: GarageFinish, dtype: int64
TA     2604
NaN     159
Fa      124
Gd       24
Po        5
Ex        3
Name: GarageQual, dtype: int64
TA     2654
NaN     159
Fa       74
Gd       15
Po       14
Ex        3
Name: GarageCond, dtype: int64
```

In [20]:

```
Basement_list=[col for col in objectcols if col.startswith("Bsm")]
Basement_list
```

Out[20]:

```
['BsmtQual', 'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinType2']
```

In [21]:

```
for col in Basement_list:
    freq=objectcols[col].value_counts(dropna=False)
    print(freq)
```

```
TA     1283
Gd     1209
Ex      258
Fa       88
NaN      81
Name: BsmtQual, dtype: int64
TA     2606
Gd      122
Fa      104
NaN      82
Po        5
Name: BsmtCond, dtype: int64
No     1904
Av      418
Gd      276
Mn      239
NaN      82
Name: BsmtExposure, dtype: int64
Unf    851
GLQ    849
ALQ    429
Rec    288
BLQ    269
LwQ    154
NaN     79
Name: BsmtFinType1, dtype: int64
Unf    2493
Rec     105
LwQ      87
NaN      80
BLQ      68
ALQ      52
GLQ      34
Name: BsmtFinType2, dtype: int64
```

In [22]:

```
for col in objectcols.columns:
    objectcols[col]=objectcols[col].fillna(
        objectcols[col].value_counts().idxmax())
```

In [23]:

```
numericcols.isnull().sum().sort_values(ascending=False)
```

Out[23]:

```
LotFrontage      486
GarageYrBlt      159
MasVnrArea        23
BsmtHalfBath       2
BsmtFullBath       2
BsmtFinSF2         1
GarageCars         1
GarageArea         1
TotalBsmtSF        1
BsmtUnfSF          1
BsmtFinSF1         1
KitchenAbvGr       0
3SsnPorch          0
EnclosedPorch      0
OpenPorchSF        0
WoodDeckSF         0
ScreenPorch        0
PoolArea           0
MiscVal            0
MoSold             0
Fireplaces         0
TotRmsAbvGrd       0
Id                 0
BedroomAbvGr       0
HalfBath           0
FullBath           0
MSSubClass         0
GrLivArea          0
LowQualFinSF       0
2ndFlrSF           0
1stFlrSF           0
YearRemodAdd       0
YearBuilt          0
OverallCond        0
OverallQual        0
LotArea            0
YrSold             0
dtype: int64
```

In [24]:

```
numericcols.columns
```

Out[24]:

```
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'],
      dtype='object')
```

In [25]:

```
categorycols=numericcols[['OverallQual','OverallCond','YearBuilt','YearRemodAdd','GarageYrBlt','MoSold','YrSold']]
```

In [26]:

```
numericcols=numericcols.drop(['OverallQual','OverallCond','YearBuilt','YearRemodAdd','GarageYrBlt','MoSold','YrSold'],axis=1)
```

In [27]:

```
print(categorycols.shape)
print(numericcols.shape)
```

```
(2919, 7)
(2919, 30)
```

In [28]:

```
categorycols.isnull().sum().sort_values(ascending=False)
```

Out[28]:

```
GarageYrBlt     159
OverallQual       0
OverallCond       0
YearBuilt         0
YearRemodAdd      0
MoSold            0
YrSold            0
dtype: int64
```

In [29]:

```
categorycols.GarageYrBlt=categorycols.GarageYrBlt.fillna(categorycols.GarageYrBlt.value_counts().idxmax())
```

In [30]:

```
# Median Imputation for Numeric Cols
for col in numericcols.columns:
    numericcols[col]=numericcols[col].fillna(numericcols[col].median())
```

In [31]:

```
from sklearn.preprocessing import LabelEncoder
```

In [32]:

```
numericcols["SalePrice"]=objectcols.SalePrice
```

In [33]:

```
objectcols=objectcols.drop("SalePrice",axis=1)
```

In [34]:

```
le=LabelEncoder()
```

In [35]:

```
objectcols_encode=objectcols.apply(le.fit_transform)
```

In [36]:

```
categorycols_encode=categorycols.apply(le.fit_transform)
```

In [37]:

```
numericcols.head()
```

Out[37]:

	Id	MSSubClass	LotFrontage	LotArea	MasVnrArea	BsmtFinSF1	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	1stFlrSF	...	GarageCars	GarageArea	Wood
0	1	60	65.0	8450	196.0	706.0	0.0	150.0	856.0	856	...	2.0	548.0	
1	2	20	80.0	9600	0.0	978.0	0.0	284.0	1262.0	1262	...	2.0	460.0	
2	3	60	68.0	11250	162.0	486.0	0.0	434.0	920.0	920	...	2.0	608.0	
3	4	70	60.0	9550	0.0	216.0	0.0	540.0	756.0	961	...	3.0	642.0	
4	5	60	84.0	14260	350.0	655.0	0.0	490.0	1145.0	1145	...	3.0	836.0	

5 rows × 31 columns

In [38]:

```
# Check for Multicollinearity
numericcols.drop('SalePrice',axis=1).corr()
```

Out[38]:

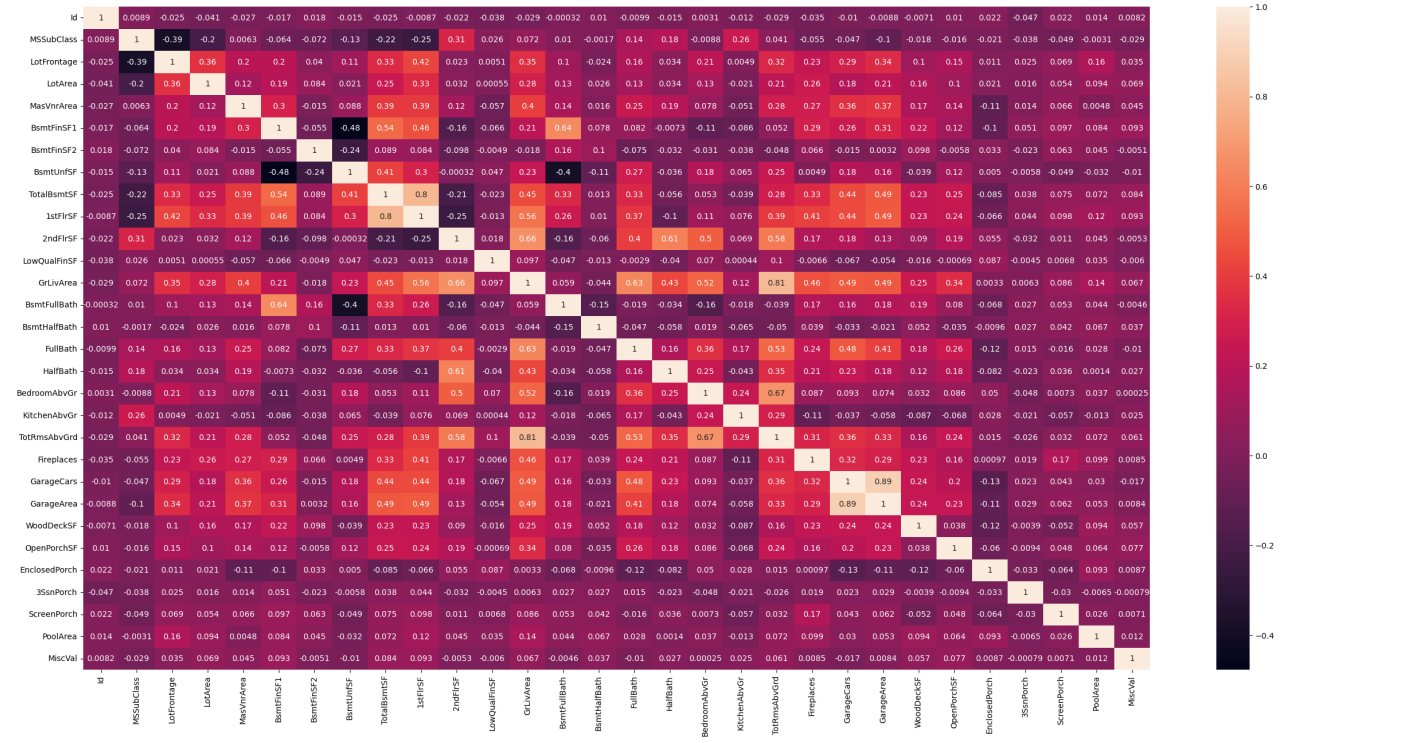
	Id	MSSubClass	LotFrontage	LotArea	MasVnrArea	BsmtFinSF1	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	1stFlrSF	...	Fireplaces	GarageCars
Id	1.000000	0.008931	-0.024710	-0.040746	-0.026737	-0.016988	0.018170	-0.014509	-0.024960	-0.008678	...	-0.035236	-0.01006
MSSubClass	0.008931	1.000000	-0.389469	-0.201730	0.006309	-0.064254	-0.072431	-0.125913	-0.219893	-0.248641	...	-0.055151	-0.04656
LotFrontage	-0.024710	-0.389469	1.000000	0.361426	0.200474	0.201697	0.040330	0.106921	0.330765	0.423217	...	0.231731	0.29063
LotArea	-0.040746	-0.201730	0.361426	1.000000	0.124728	0.194050	0.084107	0.021400	0.254150	0.332460	...	0.261185	0.18041
MasVnrArea	-0.026737	0.006309	0.200474	0.124728	1.000000	0.301427	-0.014580	0.087991	0.393662	0.392367	...	0.273129	0.35765
BsmtFinSF1	-0.016988	-0.064254	0.201697	0.194050	0.301427	1.000000	-0.055028	-0.477387	0.536471	0.458091	...	0.293095	0.25551
BsmtFinSF2	0.018170	-0.072431	0.040330	0.084107	-0.014580	-0.055028	1.000000	-0.238215	0.089423	0.084389	...	0.065707	-0.01475
BsmtUnfSF	-0.014509	-0.125913	0.106921	0.021400	0.087991	-0.477387	-0.238215	1.000000	0.412291	0.296623	...	0.004881	0.18006
TotalBsmtSF	-0.024960	-0.219893	0.330765	0.254150	0.393662	0.536471	0.089423	0.412291	1.000000	0.801638	...	0.332948	0.43790

In [39]:

```
import seaborn as sns
```

In [40]:

```
plt.figure(figsize=(30,15))
sns.heatmap(numericcols.drop('SalePrice',axis=1).corr(),annot=True)
plt.show()
```



In [41]:

```
from sklearn.preprocessing import StandardScaler
```

In [42]:

```
scaler=StandardScaler()
```

In [43]:

```
numericcols_scaled=scaler.fit_transform(numericcols.drop(
'SalePrice',axis=1))
```

In [44]:

```
numericcols_scaled=pd.DataFrame(numericcols_scaled,
columns=numericcols.columns[0:30])
```

In [45]:

```
numericcols_scaled=numericcols_scaled.reset_index()
```

In [46]:

```
objectcols_encode=objectcols_encode.reset_index()
```

In [47]:

```
categorycols_encode=categorycols_encode.reset_index()
```

In [48]:

```
combinedf_clean=pd.concat([numericcols_scaled,objectcols_encode,
categorycols_encode],axis=1)
```

In [49]:

```
numericcols=numericcols.reset_index()
```

In [50]:

```
combinedf_clean['SalePrice']=numericcols.SalePrice
```

In [51]:

```
housetrain_df=combinedf_clean[combinedf_clean.SalePrice!='test']
housetest_df=combinedf_clean[combinedf_clean.SalePrice=='test']
```

In [52]:

```
housetest_df=housetest_df.drop('SalePrice',axis=1)
```

In [53]:

```
# Split data into X and Y
y=housetrain_df.SalePrice
x=housetrain_df.drop(['Id','SalePrice','index'],axis=1)
```

In [54]:

```
housetest_df=housetest_df.drop(['Id','index'],axis=1)
```

In [55]:

```
housetest_df.head()
```

Out[55]:

	MSSubClass	LotFrontage	LotArea	MasVnrArea	BsmtFinSF1	BsmtFinSF2	BsmtUnfSF	TotalBsmtSF	1stFlrSF	2ndFlrSF	...	MiscFeature	SaleTy
1460	-0.873616	0.511940	0.184371	-0.567016	0.058407	0.558290	-0.661680	-0.385270	-0.671897	-0.785025	...	4	
1461	-0.873616	0.558857	0.519791	0.036930	1.057404	-0.293025	-0.352165	0.629217	0.431865	-0.785025	...	0	
1462	0.067331	0.230438	0.464374	-0.567016	0.767585	-0.293025	-0.964368	-0.280871	-0.590326	0.850426	...	4	
1463	0.067331	0.418106	-0.024109	-0.455174	0.352617	-0.293025	-0.538784	-0.285410	-0.595424	0.796766	...	4	
1464	1.478753	-1.223988	-0.654748	-0.567016	-0.391691	-0.293025	1.038379	0.518010	0.306959	-0.785025	...	4	

5 rows × 79 columns

In [56]:

```
y.dtype
```

Out[56]:

```
dtype('O')
```

In [57]:

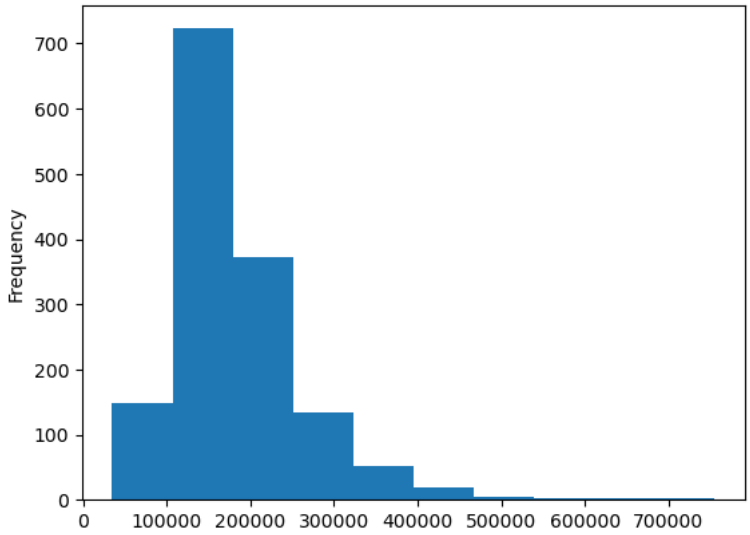
```
y=y.astype('int64')
```

In [58]:

```
y.plot(kind="hist")
```

Out[58]:

```
<AxesSubplot:ylabel='Frequency'>
```

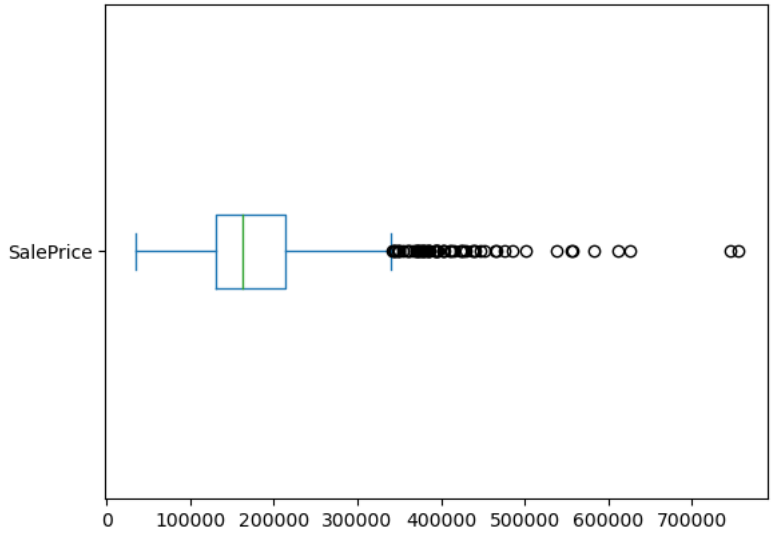


In [59]:

```
y.plot(kind="box",vert=False)
```

Out[59]:

```
<AxesSubplot:>
```

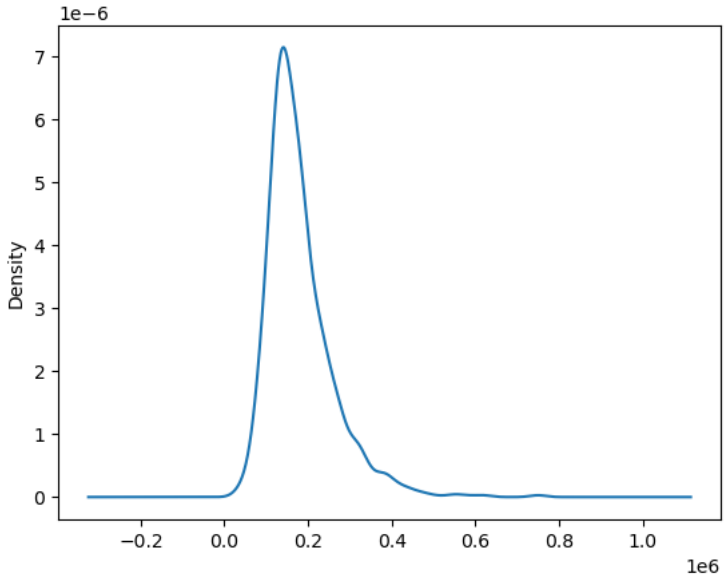


In [60]:

```
y.plot(kind="density")
```

Out[60]:

```
<AxesSubplot:ylabel='Density'>
```



In [61]:

```
from sklearn.linear_model import LinearRegression
```

In [62]:

```
reg=LinearRegression()
```

In [63]:

```
regmodel=reg.fit(x,y)
```

In [64]:

```
regmodel.score(x,y)
```

Out[64]:

```
0.8545634224507179
```

In [65]:

```
regtestpredict=regmodel.predict(housetest_df)
```

In [66]:

```
regtestpredict
```

Out[66]:

```
array([106189.76326881, 156987.78817192, 167242.08767411, ...,
       143662.33304422, 115297.31625419, 242815.16584994])
```

In [67]:

```
pd.DataFrame(regtestpredict).to_csv("reg.csv")
```

In [ ]:

```

```