In [2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

In [7]:
```
data = pd.read_table('HousePrices.csv',sep=',')
data
```

Out[7]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondi
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2008	WD	Nor
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5	2007	WD	Nor
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	9	2008	WD	Nor
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2006	WD	Abno
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	12	2008	WD	Nor
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1455	1456	60	RL	62.0	7917	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	8	2007	WD	Nor
1456	1457	20	RL	85.0	13175	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	2	2010	WD	Nor
1457	1458	70	RL	66.0	9042	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	GdPrv	Shed	2500	5	2010	WD	Nor
1458	1459	20	RL	68.0	9717	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	4	2010	WD	Nor
1459	1460	20	RL	75.0	9937	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	6	2008	WD	Nor

1460 rows × 81 columns

In [8]:
```
data.shape
```

Out[8]:
```
(1460, 81)
```

In [9]:
```
data.head()
```

Out[9]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	S
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2008	WD	Normal	
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5	2007	WD	Normal	
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	9	2008	WD	Normal	
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	2	2006	WD	Abnorml	
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	12	2008	WD	Normal	

5 rows × 81 columns

In [10]:
```
data.columns
```

Out[10]:
```
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 [11]:
```
data.dtypes
```

Out[11]:
```
Id               int64
MSSubClass       int64
MSZoning        object
LotFrontage    float64
LotArea          int64
                ...
MoSold           int64
YrSold           int64
SaleType        object
SaleCondition   object
SalePrice       int64
Length: 81, dtype: object
```

In [12]:
```
data.tail()
```

Out[12]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondi
1455	1456	60	RL	62.0	7917	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	8	2007	WD	Nor
1456	1457	20	RL	85.0	13175	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	2	2010	WD	Nor
1457	1458	70	RL	66.0	9042	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	GdPrv	Shed	2500	5	2010	WD	Nor
1458	1459	20	RL	68.0	9717	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	4	2010	WD	Nor
1459	1460	20	RL	75.0	9937	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	6	2008	WD	Nor

5 rows × 81 columns

In [13]:
```
data.sample(10)
```

Out[13]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondi
1356	1357	20	RL	NaN	9477	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	GdWo	NaN	0	10	2008	WD	Nor
380	381	50	RL	50.0	5000	Pave	Pave	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5	2010	WD	Nor
244	245	60	RL	NaN	8880	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5	2010	WD	Nor
303	304	20	RL	75.0	9800	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	GdWo	NaN	0	7	2006	WD	Abno
940	941	90	RL	55.0	12640	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	7	2006	WD	Nor
444	445	60	RL	70.0	8750	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	7	2008	WD	Nor
526	527	20	RL	70.0	13300	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	6	2007	WD	Nor
530	531	80	RL	85.0	10200	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	8	2008	WD	Abno
465	466	120	RM	NaN	3072	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0	5	2006	WD	Nor
1017	1018	120	RL	NaN	5814	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	8	2009	COD	Abno

10 rows × 81 columns

In [14]:
```
data.info()
```

Out[14]:
```
<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 [15]:
```
data.describe()
```

Out[15]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	YearRemodAdd	MasVnrArea	BsmtFinSf1	...	WoodDeckSF	OpenPorchSF	EnclosedPorch	3S
count	1460.000000	1460.000000	1201.000000	1460.000000	1460.000000	1460.000000	1460.000000	1460.000000	1452.000000	1460.000000	...	1460.000000	1460.000000	1460.000000	146
mean	730.500000	56.897260	70.049958	10516.828082	6.099315	5.575342	1971.267808	1984.865753	103.685262	443.639726	...	94.244521	46.660274	21.954110	
std	421.610009	42.300571	24.284752	9981.264932	1.382997	1.112799	30.202904	20.645407	181.066207	456.098091	...	125.338794	66.256028	61.119149	2
min	1.000000	20.000000	21.000000	1300.000000	1.000000	1.000000	1872.000000	1950.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	
25%	365.750000	20.000000	59.000000	7553.500000	5.000000	5.000000	1954.000000	1967.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000	
50%	730.500000	50.000000	69.000000	9478.500000	6.000000	5.000000	1973.000000	1994.000000	0.000000	383.500000	...	0.000000	25.000000	0.000000	
75%	1095.250000	70.000000	80.000000	11601.500000	7.000000	6.000000	2000.000000	2004.000000	166.000000	712.250000	...	168.000000	68.000000	0.000000	
max	1460.000000	190.000000	313.000000	215245.000000	10.000000	9.000000	2010.000000	2010.000000	1600.000000	5644.000000	...	857.000000	547.000000	552.000000	50

8 rows × 38 columns

In [23]:
```
data.isnull().sum()
```

Out[23]:
```
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 [26]:
```
median=data['LotFrontage'].median()
data['LotFrontage'].replace(np.nan,median,inplace=True)
```

In [27]:
```
data.isnull().sum()
```

Out[27]:
```
Id             0
MSSubClass     0
MSZoning       0
LotFrontage    0
LotArea        0
              ..
MoSold         0
YrSold         0
SaleType       0
SaleCondition  0
SalePrice      0
Length: 81, dtype: int64
```

In [29]:
```
duplicate=data.duplicated()
print(duplicate.sum())
data[duplicate]
```

Out[29]:
```
0
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	Sa

0 rows × 81 columns

In [30]:
```
data.boxplot(column=['LotFrontage'])
```

Out[30]:
```
<AxesSubplot:>
```

In [32]:
```
data=data.drop('Street',axis=1)
```

In [33]:
```
data.head()
```

Out[33]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Alley	LotShape	LandContour	Utilities	LotConfig	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition
0	1	60	RL	65.0	8450	NaN	Reg	Lvl	AllPub	FR2	...	0	NaN	NaN	NaN	0	2	2008	WD	Norma
1	2	20	RL	80.0	9600	NaN	Reg	Lvl	AllPub	FR2	...	0	NaN	NaN	NaN	0	5	2007	WD	Norma
2	3	60	RL	68.0	11250	NaN	IR1	Lvl	AllPub	Inside	...	0	NaN	NaN	NaN	0	9	2008	WD	Norma
3	4	70	RL	60.0	9550	NaN	IR1	Lvl	AllPub	Corner	...	0	NaN	NaN	NaN	0	2	2006	WD	Abnorm
4	5	60	RL	84.0	14260	NaN	IR1	Lvl	AllPub	FR2	...	0	NaN	NaN	NaN	0	12	2008	WD	Norma

5 rows × 80 columns

In [35]:
```
data=data.drop(['Alley','LotShape','LandContour','Utilities','LotConfig','PoolArea','PoolQC','Fence','MiscVal'],axis=1)
data.head()
```

Out[35]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	LandSlope	Neighborhood	Condition1	Condition2	BldgType	...	OpenPorchSF	EnclosedPorch	3SsnPorch	ScreenPorch	MiscFeature	MoSold
0	1	60	RL	65.0	8450	Gtl	CollgCr	Norm	Norm	1Fam	...	61	0	0	0	NaN	2
1	2	20	RL	80.0	9600	Gtl	Veenker	Feedr	Norm	1Fam	...	0	0	0	0	NaN	5
2	3	60	RL	68.0	11250	Gtl	CollgCr	Norm	Norm	1Fam	...	42	0	0	0	NaN	9
3	4	70	RL	60.0	9550	Gtl	Crawfor	Norm	Norm	1Fam	...	35	272	0	0	NaN	2
4	5	60	RL	84.0	14260	Gtl	NoRidge	Norm	Norm	1Fam	...	84	0	0	0	NaN	12

5 rows × 71 columns

In [36]:
```
data=pd.get_dummies(data,columns=['SaleCondition'])
data.head()
```

Out[36]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	LandSlope	Neighborhood	Condition1	Condition2	BldgType	...	MoSold	YrSold	SaleType	SalePrice	SaleCondition_Abnorml	SaleCondition_Adj
0	1	60	RL	65.0	8450	Gtl	CollgCr	Norm	Norm	1Fam	...	2	2008	WD	208500	0	
1	2	20	RL	80.0	9600	Gtl	Veenker	Feedr	Norm	1Fam	...	5	2007	WD	181500	0	
2	3	60	RL	68.0	11250	Gtl	CollgCr	Norm	Norm	1Fam	...	9	2008	WD	223500	0	
3	4	70	RL	60.0	9550	Gtl	Crawfor	Norm	Norm	1Fam	...	2	2006	WD	140000	1	
4	5	60	RL	84.0	14260	Gtl	NoRidge	Norm	Norm	1Fam	...	12	2008	WD	250000	0	

5 rows × 76 columns

In [39]:
```
x=data[['YearBuilt','BedroomAbvGr','KitchenAbvGr','GarageArea','MoSold','YrSold','GarageCars','MSSubClass','LotArea']]
y=x=data['SalePrice']
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.30)
model=LinearRegression()
model.fit(x_train,y_train)
```

Out[39]:
```
LinearRegression()
```

In [40]:
```
model.score(x_train,y_train)
```

Out[40]:
```
0.5302716742245762
```

In [41]:
```
model.score(x_test,y_test)
```

Out[41]:
```
0.5045965167794824
```

In [42]:
```
from sklearn.preprocessing import PolynomialFeatures
from sklearn import linear_model
poly=PolynomialFeatures(degree=2,interaction_only=True)
x_train=poly.fit_transform(x_train)
x_test=poly.fit_transform(x_test)
poly_clf=linear_model.LinearRegression()
poly_clf.fit(x_train,y_train)
y_pred=poly_clf.predict(x_test)
print(poly_clf.score(x_train,y_train))
```

```
0.6545422603312583
```

```
print(poly_clf.score(x_test,y_test))
```

```
0.4895711058974403
```

In [ ]: