CSCI 3220: Python for Data Science and AI

Assignment 2

Due Date: March 17(Wednesday), 11:59 PM

Total Points: 16X2 = 32

Bonus Points: 5X2 = 10

***Write the code in Jupyter notebook (ipynb file) with proper comments.

***Add proper citation if you take help from a different source (not from the textbook).

***Rename the file with your student ID and submit it in Moodle.

***(No. of lines : #) shows the minimum number of lines required to do the corresponding task.

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import impute
import numpy as np
```

1. Read the dataset (CSV file) and create a DataFrame. (No. of lines : 1)

In [2]:
```

```

2. Display the first five rows of the DataFrame. (No. of lines : 1)

In [3]:
```

```

Out[3]:

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

5 rows × 81 columns

3. Print the number of rows and columns.(No. of lines : 1)

In [4]:

Out[4]: (1460, 81)

4. Display the name of columns/features, Non-Null Count, Dtype. (No. of lines : 1)

In [5]:

```
<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
```

5. Print the overall statistics for each column in the DataFrame. (No. of lines : 1)

In [6]:

Out[6]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	Y
count	1460.000000	1460.000000	1201.000000	1460.000000	1460.000000	1460.000000	1460.000000	
mean	730.500000	56.897260	70.049958	10516.828082	6.099315	5.575342	1971.267808	
std	421.610009	42.300571	24.284752	9981.264932	1.382997	1.112799	30.202904	
min	1.000000	20.000000	21.000000	1300.000000	1.000000	1.000000	1872.000000	
25%	365.750000	20.000000	59.000000	7553.500000	5.000000	5.000000	1954.000000	
50%	730.500000	50.000000	69.000000	9478.500000	6.000000	5.000000	1973.000000	

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	Y
75%	1095.250000	70.000000	80.000000	11601.500000	7.000000	6.000000	2000.000000	
max	1460.000000	190.000000	313.000000	215245.000000	10.000000	9.000000	2010.000000	

8 rows × 38 columns

6. Print the statistics for the “SalePrice” column.(No. of lines : 1)

In [7]:

```
count      1460.000000
mean     180921.195890
std       79442.502883
min       34900.000000
25%      129975.000000
50%      163000.000000
75%      214000.000000
max      755000.000000
Name: SalePrice, dtype: float64
```

7. Print the name of the columns.(No. of lines : 1)

In [8]:

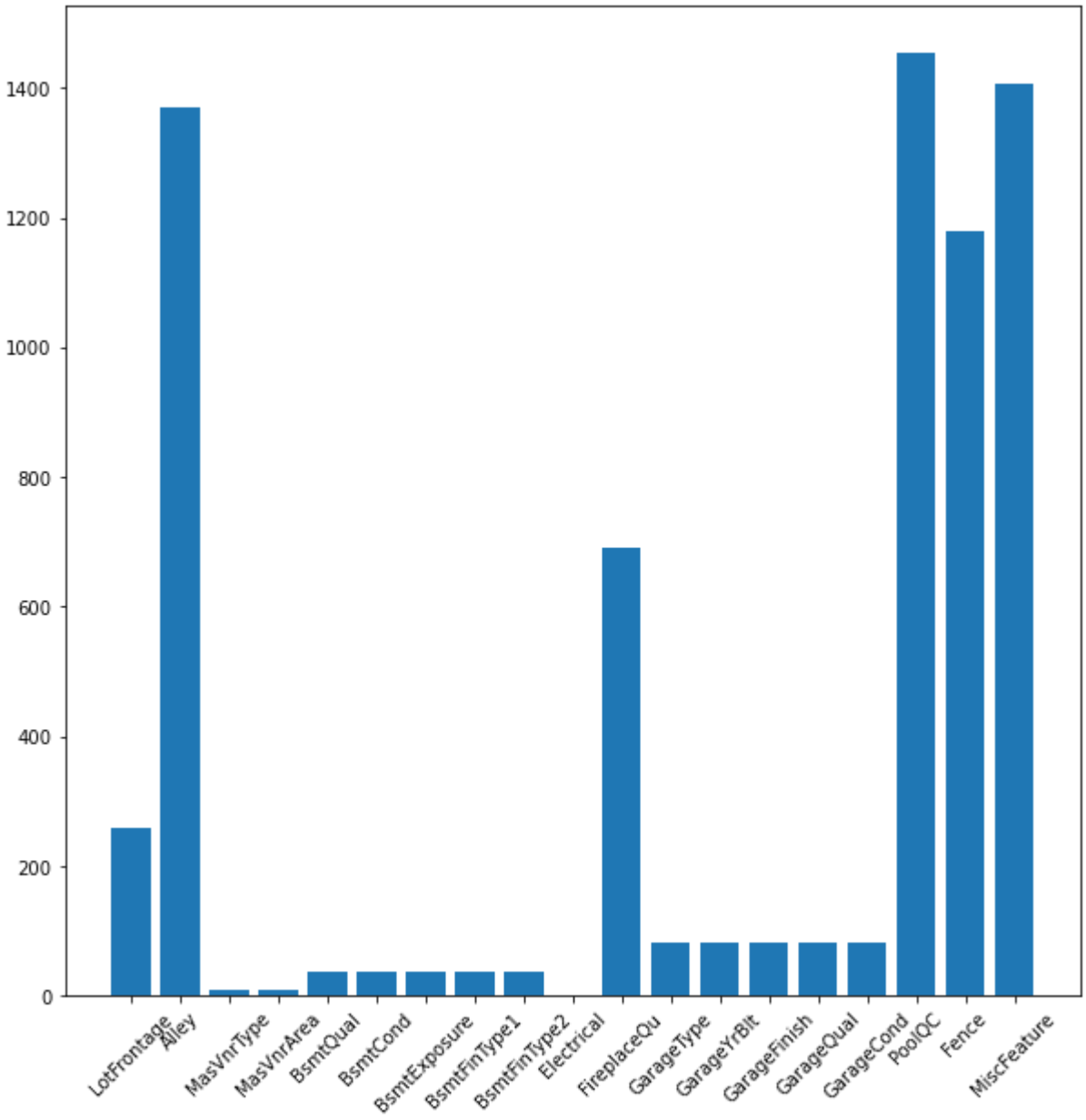
```
Index(['Id', 'MSSubClass', 'MSZoning', 'LotFrontage', 'LotArea', 'Street',
       'Alley', 'LotShape', 'LandContour', 'Utilities', 'LotConfig',
       'LandSlope', 'Neighborhood', 'Condition1', 'Condition2', 'BldgType',
       'HouseStyle', 'OverallQual', 'OverallCond', 'YearBuilt', 'YearRemodAdd',
       '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')
```

8. Print the number of missing values for each column and draw a figure to display the number of missing values for each column. (No. of lines : 7)

In [9]:

```
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
```



9. Create two new DataFrames. One for Categorical features and one for numerical features. For each

DataFrame, print the name of the columns and DataFrame . (No. of lines : 6)

In [10]:

```
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')
```

Out[10]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	YearRemodAdd	MasVnr
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 [11]:

```
Index(['MSZoning', 'Street', 'Alley', 'LotShape', 'LandContour', 'Utilities',
       'LotConfig', 'LandSlope', 'Neighborhood', 'Condition1', 'Condition2',
       'BldgType', 'HouseStyle', 'RoofStyle', 'RoofMatl', 'Exterior1st',
       'Exterior2nd', 'MasVnrType', 'ExterQual', 'ExterCond', 'Foundation',
       'BsmtQual', 'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinType2',
       'Heating', 'HeatingQC', 'CentralAir', 'Electrical', 'KitchenQual',
       'Functional', 'FireplaceQu', 'GarageType', 'GarageFinish', 'GarageQual',
       'GarageCond', 'PavedDrive', 'PoolQC', 'Fence', 'MiscFeature',
       'SaleType', 'SaleCondition'],
      dtype='object')
```

Out[11]:

	MSZoning	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood	Co
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	

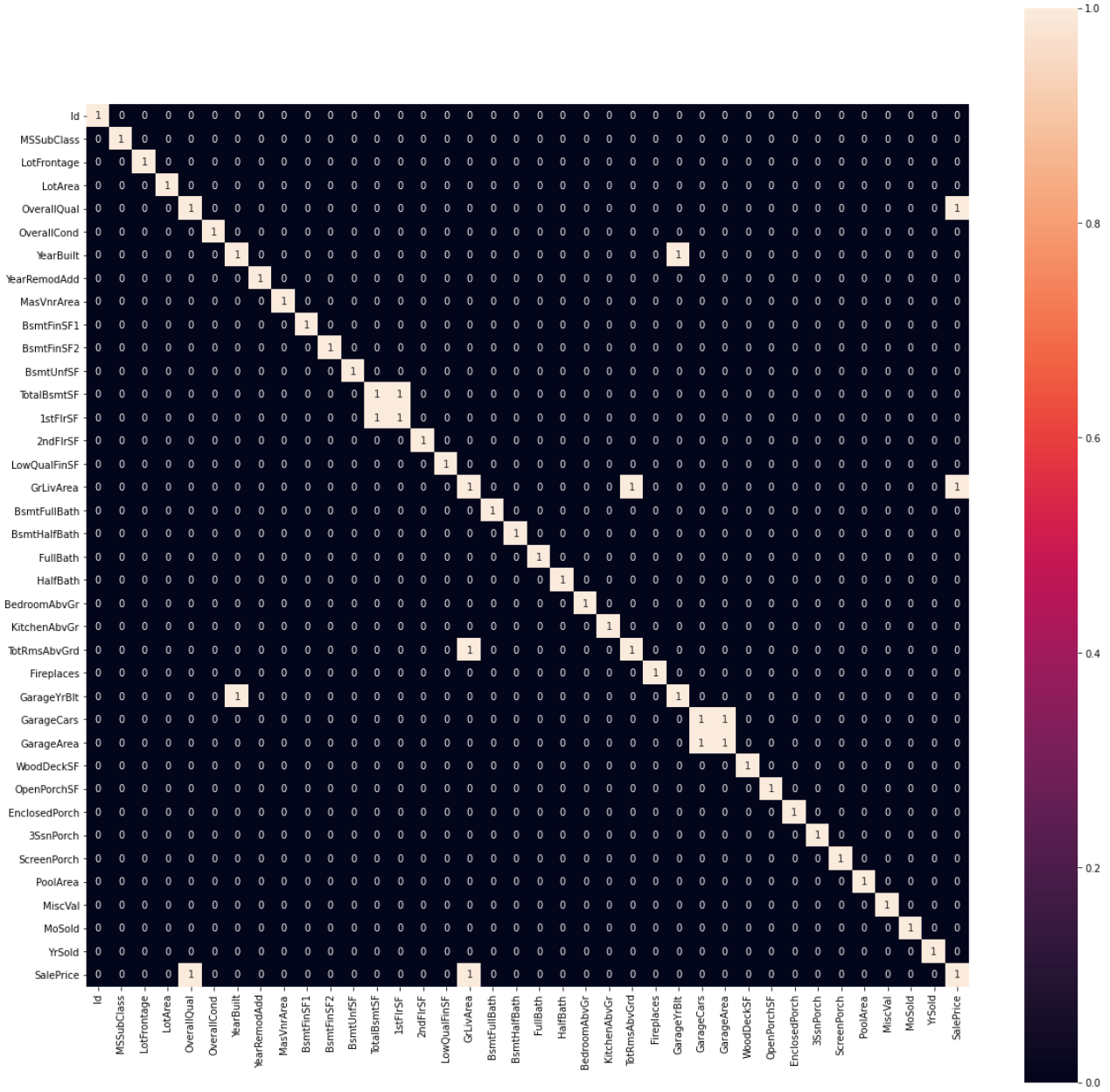
5 rows × 43 columns

10. Calculate the correlation for the pair of numerical features. Draw a heatmap diagram to display only the

pairs where the correlation is greater than 0.8. Encode the result as 0 (corr<0.7) and 1 (corr>0.7). (No. of lines : 3)

In [12]:

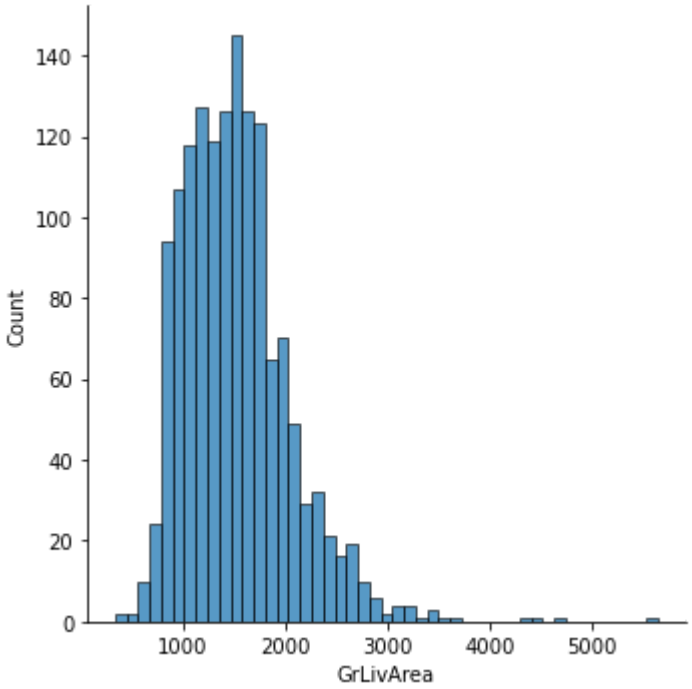
Out[12]: <AxesSubplot:>



11. Draw a histogram plot for feature “GrLivArea” (No. of lines : 1)

In [13]:

Out[13]: <seaborn.axisgrid.FacetGrid at 0x29f34246ca0>

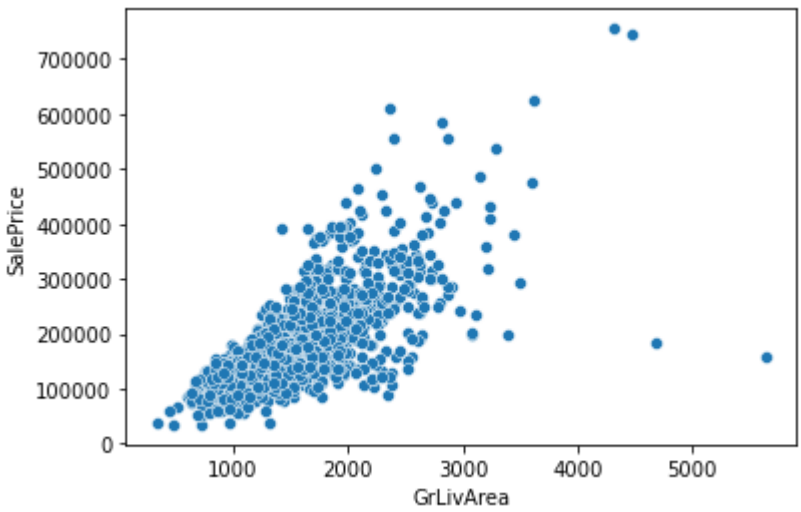


12. Draw a scatter plot for feature “GrLivArea” on the x-axis and “SalePrice” on the y-axis. (No. of lines : 1)

In [14]:

```
F:\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the follow
ing variables as keyword args: x, y. From version 0.12, the only valid positional argume
nt will be `data`, and passing other arguments without an explicit keyword will result i
n an error or misinterpretation.
  warnings.warn(
```

Out[14]: <AxesSubplot:xlabel='GrLivArea', ylabel='SalePrice'>

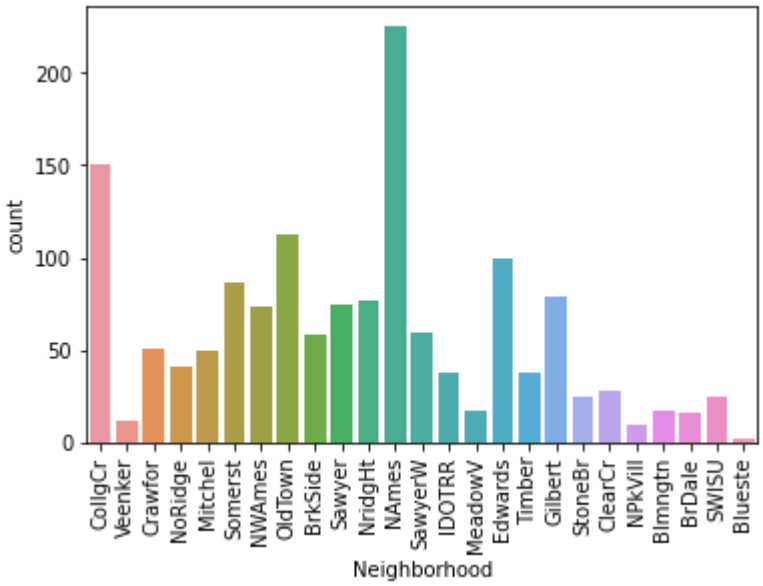


13. Draw a count plot for feature “Neighborhood” (No. of lines : 1)

In [15]:

```
F:\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the follow
ing variable as a keyword arg: x. From version 0.12, the only valid positional argument
will be `data`, and passing other arguments without an explicit keyword will result in a
n error or misinterpretation.
  warnings.warn(
```

Out[15]: <AxesSubplot:xlabel='Neighborhood', ylabel='count'>



14. Use a "SimpleImputer" method from the sci-kit learn library to impute the missing values in the DataFrame with numerical features. Print the first five rows in DataFrame.

https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html (No. of lines : 4)

In [16]:

Out[16]:

	Id	MSSubClass	LotFrontage	LotArea	OverallQual	OverallCond	YearBuilt	YearRemodAdd	MasVn
0	1.0	60.0	65.0	8450.0	7.0	5.0	2003.0	2003.0	
1	2.0	20.0	80.0	9600.0	6.0	8.0	1976.0	1976.0	
2	3.0	60.0	68.0	11250.0	7.0	5.0	2001.0	2002.0	
3	4.0	70.0	60.0	9550.0	7.0	5.0	1915.0	1970.0	
4	5.0	60.0	84.0	14260.0	8.0	5.0	2000.0	2000.0	

5 rows × 38 columns

15. For categorical features, fill up the missing values with a string "Empty". Print the first five rows in DataFrame. (No. of lines : 2)

In [17]:

Out[17]:

	MSZoning	Street	Alley	LotShape	LandContour	Utilities	LotConfig	LandSlope	Neighborhood	C
0	RL	Pave	None	Reg	Lvl	AllPub	Inside	Gtl	CollgCr	
1	RL	Pave	None	Reg	Lvl	AllPub	FR2	Gtl	Veenker	
2	RL	Pave	None	IR1	Lvl	AllPub	Inside	Gtl	CollgCr	
3	RL	Pave	None	IR1	Lvl	AllPub	Corner	Gtl	Crawfor	
4	RL	Pave	None	IR1	Lvl	AllPub	FR2	Gtl	NoRidge	

5 rows × 43 columns

16. Concatenate the two DataFrames (categorical and numerical) to create a new DataFrame. Display the concatenated DataFrame. Print the number of missing values for each column to make sure there are no missing values. (No. of lines : 5)

In [18]:

```
     MSZoning Street Alley LotShape LandContour Utilities LotConfig LandSlope  \
0          RL   Pave  None      Reg         Lvl    AllPub    Inside       Gtl
1          RL   Pave  None      Reg         Lvl    AllPub       FR2       Gtl
2          RL   Pave  None      IR1         Lvl    AllPub    Inside       Gtl
3          RL   Pave  None      IR1         Lvl    AllPub    Corner       Gtl
4          RL   Pave  None      IR1         Lvl    AllPub       FR2       Gtl
...       ...    ...   ...      ...         ...       ...       ...       ...
1455       RL   Pave  None      Reg         Lvl    AllPub    Inside       Gtl
1456       RL   Pave  None      Reg         Lvl    AllPub    Inside       Gtl
1457       RL   Pave  None      Reg         Lvl    AllPub    Inside       Gtl
1458       RL   Pave  None      Reg         Lvl    AllPub    Inside       Gtl
1459       RL   Pave  None      Reg         Lvl    AllPub    Inside       Gtl

     Neighborhood Condition1  ... WoodDeckSF OpenPorchSF EnclosedPorch  \
0         CollgCr       Norm  ...        0.0        61.0           0.0
1         Veenker      Feedr  ...      298.0         0.0           0.0
2         CollgCr       Norm  ...        0.0        42.0           0.0
3         Crawfor       Norm  ...        0.0        35.0         272.0
4         NoRidge       Norm  ...      192.0        84.0           0.0
...           ...        ...  ...        ...         ...           ...
1455      Gilbert       Norm  ...        0.0        40.0           0.0
1456       NWAmes       Norm  ...      349.0         0.0           0.0
1457      Crawfor       Norm  ...        0.0        60.0           0.0
1458        NAmes       Norm  ...      366.0         0.0         112.0
1459      Edwards       Norm  ...      736.0        68.0           0.0

     3SsnPorch ScreenPorch PoolArea MiscVal MoSold  YrSold SalePrice
0          0.0         0.0      0.0     0.0    2.0  2008.0  208500.0
1          0.0         0.0      0.0     0.0    5.0  2007.0  181500.0
2          0.0         0.0      0.0     0.0    9.0  2008.0  223500.0
3          0.0         0.0      0.0     0.0    2.0  2006.0  140000.0
4          0.0         0.0      0.0     0.0   12.0  2008.0  250000.0
...        ...         ...      ...     ...    ...     ...       ...
1455       0.0         0.0      0.0     0.0    8.0  2007.0  175000.0
```

```
1456       0.0         0.0      0.0     0.0    2.0  2010.0  210000.0
1457       0.0         0.0      0.0  2500.0    5.0  2010.0  266500.0
1458       0.0         0.0      0.0     0.0    4.0  2010.0  142125.0
1459       0.0         0.0      0.0     0.0    6.0  2008.0  147500.0

[1460 rows x 81 columns]
Series([], dtype: int64)
```

17. (Bonus Points) Encode the categorical features using the following library:

https://scikit-learn.org/stable/modules/preprocessing.html#encoding-categorical-features

18. (Bonus Points) Create a profiling report (Html) for the given dataset using the pandas-profiling library.

https://github.com/pandas-profiling/pandas-profiling