Shellya Nur Atqiya 1903685

IMPORT DATA

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

train = pd.read_csv('/content/train.csv')
test = pd.read_csv('/content/test.csv')

#show top 5 rows
train.head(5)
```

```
   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 LotConfig LandSlope Neighborhood Condition1  \
0         Lvl    AllPub    Inside       Gtl      CollgCr       Norm
1         Lvl    AllPub       FR2       Gtl      Veenker      Feedr
2         Lvl    AllPub    Inside       Gtl      CollgCr       Norm
3         Lvl    AllPub    Corner       Gtl      Crawfor       Norm
4         Lvl    AllPub       FR2       Gtl      NoRidge       Norm

  Condition2 BldgType HouseStyle  OverallQual  OverallCond  YearBuilt
\
0       Norm     1Fam     2Story            7            5       2003

1       Norm     1Fam     1Story            6            8       1976

2       Norm     1Fam     2Story            7            5       2001

3       Norm     1Fam     2Story            7            5       1915
```

```
4       Norm     1Fam     2Story            8            5       2000


   YearRemodAdd RoofStyle RoofMatl Exterior1st Exterior2nd MasVnrType
\
0          2003     Gable  CompShg     VinylSd     VinylSd    BrkFace

1          1976     Gable  CompShg     MetalSd     MetalSd       None

2          2002     Gable  CompShg     VinylSd     VinylSd    BrkFace

3          1970     Gable  CompShg     Wd Sdng     Wd Shng       None

4          2000     Gable  CompShg     VinylSd     VinylSd    BrkFace


   MasVnrArea ExterQual ExterCond Foundation BsmtQual BsmtCond
BsmtExposure  \
0       196.0        Gd        TA      PConc       Gd       TA
No
1         0.0        TA        TA     CBlock       Gd       TA
Gd
2       162.0        Gd        TA      PConc       Gd       TA
Mn
3         0.0        TA        TA     BrkTil       TA       Gd
No
4       350.0        Gd        TA      PConc       Gd       TA
Av

  BsmtFinType1  BsmtFinSF1 BsmtFinType2  BsmtFinSF2  BsmtUnfSF
TotalBsmtSF  \
0          GLQ         706          Unf           0        150
856
1          ALQ         978          Unf           0        284
1262
2          GLQ         486          Unf           0        434
920
3          ALQ         216          Unf           0        540
756
4          GLQ         655          Unf           0        490
1145

  Heating HeatingQC CentralAir Electrical  1stFlrSF  2ndFlrSF
LowQualFinSF  \
0    GasA        Ex          Y      SBrkr       856       854
0
1    GasA        Ex          Y      SBrkr      1262         0
0
```

```
2    GasA        Ex          Y      SBrkr       920       866
0
3    GasA        Gd          Y      SBrkr       961       756
0
4    GasA        Ex          Y      SBrkr      1145      1053
0

   GrLivArea  BsmtFullBath  BsmtHalfBath  FullBath  HalfBath
BedroomAbvGr  \
0       1710             1             0         2         1
3
1       1262             0             1         2         0
3
2       1786             1             0         2         1
3
3       1717             1             0         1         0
3
4       2198             1             0         2         1
4

   KitchenAbvGr KitchenQual  TotRmsAbvGrd Functional  Fireplaces
FireplaceQu  \
0             1          Gd             8        Typ           0
NaN
1             1          TA             6        Typ           1
TA
2             1          Gd             6        Typ           1
TA
3             1          Gd             7        Typ           1
Gd
4             1          Gd             9        Typ           1
TA

  GarageType  GarageYrBlt GarageFinish  GarageCars  GarageArea
GarageQual  \
0     Attchd       2003.0          RFn           2         548
TA
1     Attchd       1976.0          RFn           2         460
TA
2     Attchd       2001.0          RFn           2         608
TA
3     Detchd       1998.0          Unf           3         642
TA
4     Attchd       2000.0          RFn           3         836
TA

  GarageCond PavedDrive  WoodDeckSF  OpenPorchSF  EnclosedPorch
3SsnPorch  \
0        TA          Y           0           61              0
0
```

```
1         TA          Y         298            0              0
0
2         TA          Y           0           42              0
0
3         TA          Y           0           35            272
0
4         TA          Y         192           84              0
0

   ScreenPorch  PoolArea PoolQC Fence MiscFeature  MiscVal  MoSold
YrSold  \
0            0         0    NaN   NaN         NaN        0       2
2008
1            0         0    NaN   NaN         NaN        0       5
2007
2            0         0    NaN   NaN         NaN        0       9
2008
3            0         0    NaN   NaN         NaN        0       2
2006
4            0         0    NaN   NaN         NaN        0      12
2008

  SaleType SaleCondition  SalePrice
0       WD        Normal     208500
1       WD        Normal     181500
2       WD        Normal     223500
3       WD       Abnorml     140000
4       WD        Normal     250000
```

```
#drop column "Id"
train = train.drop(["Id"], axis=1)
```

```
#count rows and columns
train.shape
```

```
(1460, 80)
```

```
#show columns
train.columns
```

```
Index(['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')
```

Handling Missing Values

```
train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1460 entries, 0 to 1459
Data columns (total 80 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   MSSubClass     1460 non-null   int64  
 1   MSZoning       1460 non-null   object 
 2   LotFrontage    1201 non-null   float64
 3   LotArea        1460 non-null   int64  
 4   Street         1460 non-null   object 
 5   Alley          91 non-null     object 
 6   LotShape       1460 non-null   object 
 7   LandContour    1460 non-null   object 
 8   Utilities      1460 non-null   object 
 9   LotConfig      1460 non-null   object 
 10  LandSlope      1460 non-null   object 
 11  Neighborhood   1460 non-null   object 
 12  Condition1     1460 non-null   object 
 13  Condition2     1460 non-null   object 
 14  BldgType       1460 non-null   object 
 15  HouseStyle     1460 non-null   object 
 16  OverallQual    1460 non-null   int64  
 17  OverallCond    1460 non-null   int64  
 18  YearBuilt      1460 non-null   int64  
 19  YearRemodAdd   1460 non-null   int64  
 20  RoofStyle      1460 non-null   object 
 21  RoofMatl       1460 non-null   object 
 22  Exterior1st    1460 non-null   object 
```

```
23  Exterior2nd    1460 non-null   object
24  MasVnrType     1452 non-null   object
25  MasVnrArea     1452 non-null   float64
26  ExterQual      1460 non-null   object
27  ExterCond      1460 non-null   object
28  Foundation     1460 non-null   object
29  BsmtQual       1423 non-null   object
30  BsmtCond       1423 non-null   object
31  BsmtExposure   1422 non-null   object
32  BsmtFinType1   1423 non-null   object
33  BsmtFinSF1     1460 non-null   int64
34  BsmtFinType2   1422 non-null   object
35  BsmtFinSF2     1460 non-null   int64
36  BsmtUnfSF      1460 non-null   int64
37  TotalBsmtSF    1460 non-null   int64
38  Heating        1460 non-null   object
39  HeatingQC      1460 non-null   object
40  CentralAir     1460 non-null   object
41  Electrical     1459 non-null   object
42  1stFlrSF       1460 non-null   int64
43  2ndFlrSF       1460 non-null   int64
44  LowQualFinSF   1460 non-null   int64
45  GrLivArea      1460 non-null   int64
46  BsmtFullBath   1460 non-null   int64
47  BsmtHalfBath   1460 non-null   int64
48  FullBath       1460 non-null   int64
49  HalfBath       1460 non-null   int64
50  BedroomAbvGr   1460 non-null   int64
51  KitchenAbvGr   1460 non-null   int64
52  KitchenQual    1460 non-null   object
53  TotRmsAbvGrd   1460 non-null   int64
54  Functional     1460 non-null   object
55  Fireplaces     1460 non-null   int64
56  FireplaceQu    770 non-null    object
57  GarageType     1379 non-null   object
58  GarageYrBlt    1379 non-null   float64
59  GarageFinish   1379 non-null   object
60  GarageCars     1460 non-null   int64
61  GarageArea     1460 non-null   int64
62  GarageQual     1379 non-null   object
63  GarageCond     1379 non-null   object
64  PavedDrive     1460 non-null   object
65  WoodDeckSF     1460 non-null   int64
66  OpenPorchSF    1460 non-null   int64
67  EnclosedPorch  1460 non-null   int64
68  3SsnPorch      1460 non-null   int64
69  ScreenPorch    1460 non-null   int64
70  PoolArea       1460 non-null   int64
71  PoolQC         7 non-null      object
72  Fence          281 non-null    object
```

```
 73  MiscFeature    54 non-null     object
 74  MiscVal        1460 non-null   int64
 75  MoSold         1460 non-null   int64
 76  YrSold         1460 non-null   int64
 77  SaleType       1460 non-null   object
 78  SaleCondition  1460 non-null   object
 79  SalePrice      1460 non-null   int64
dtypes: float64(3), int64(34), object(43)
memory usage: 912.6+ KB
```

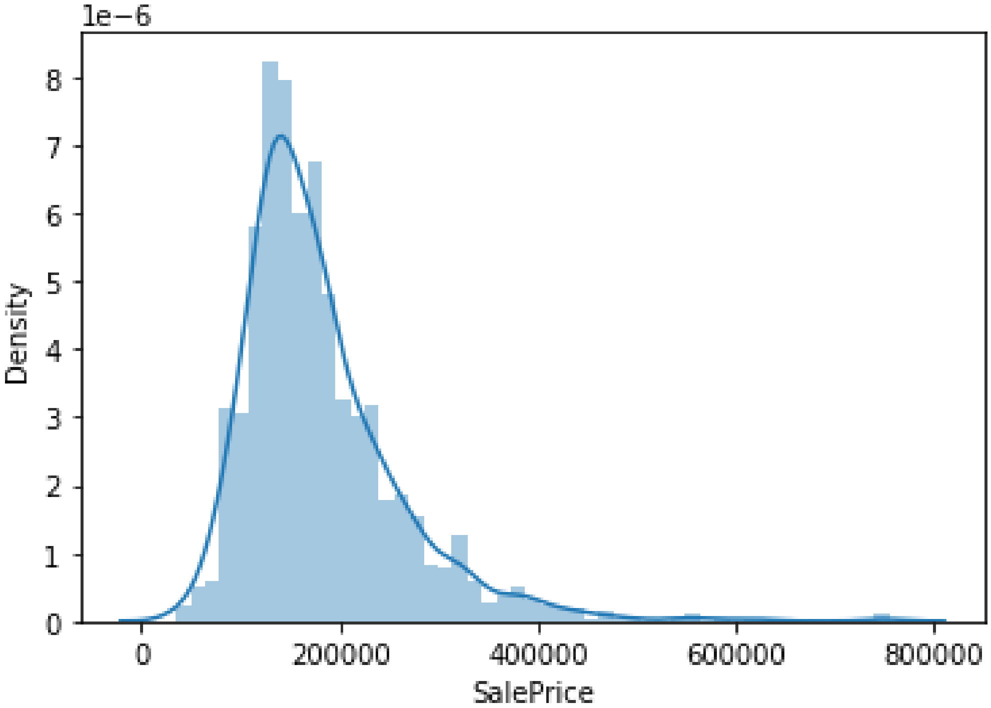
```
train["SalePrice"].describe()
```

```
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
```

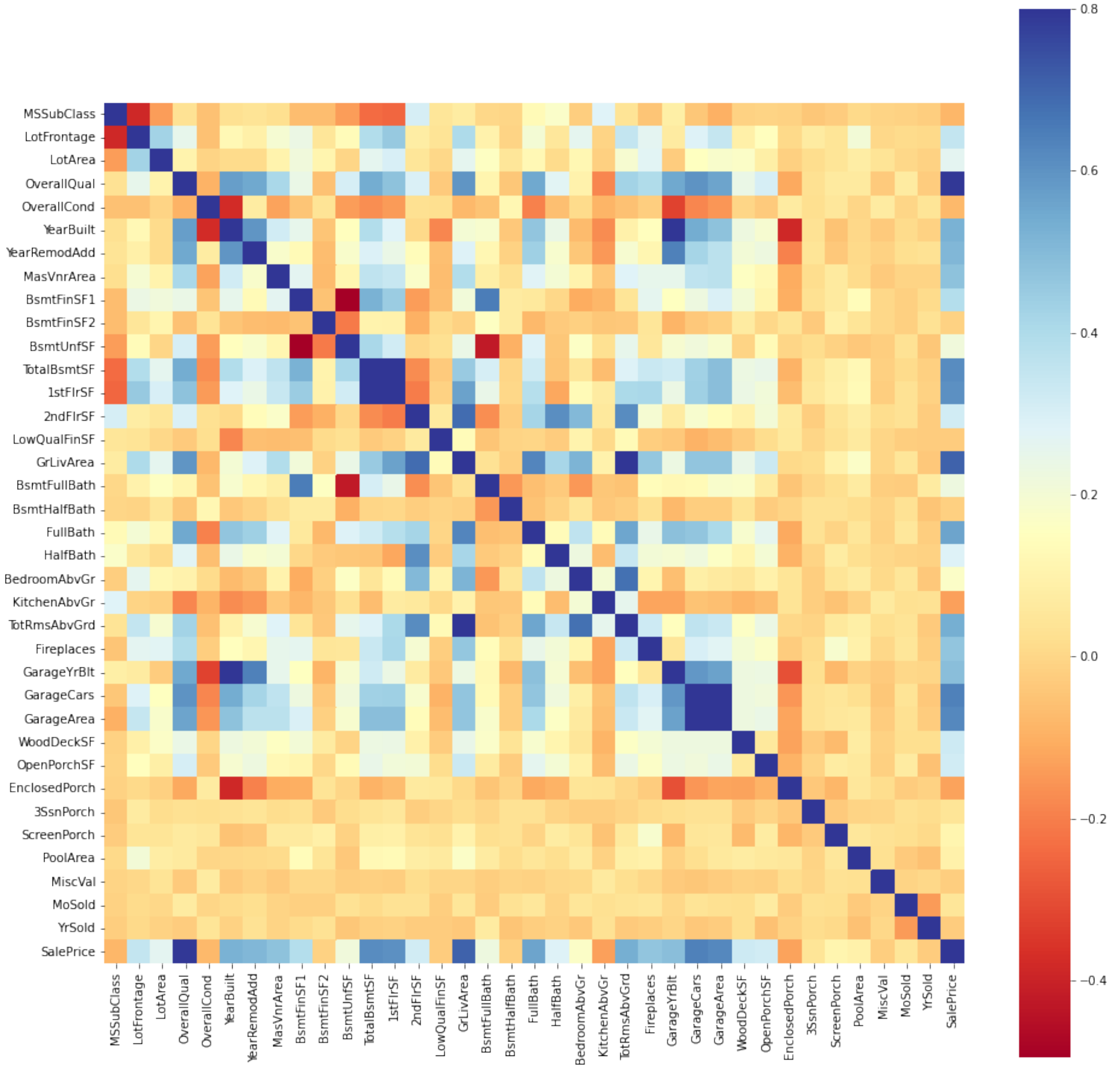
```
sns.distplot(train["SalePrice"])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed
in a future version. Please adapt your code to use either `displot` (a
figure-level function with similar flexibility) or `histplot` (an
axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

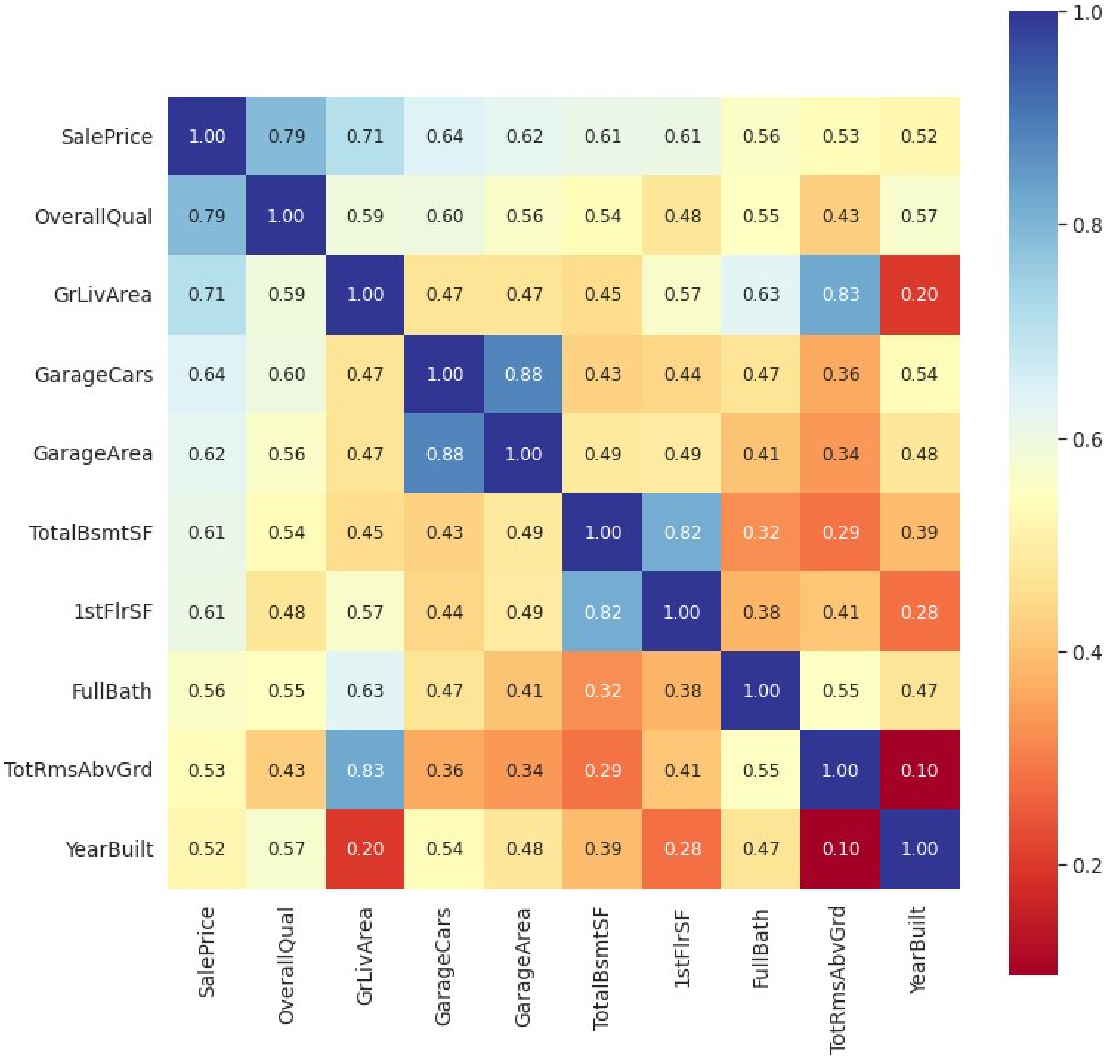
```
<matplotlib.axes._subplots.AxesSubplot at 0x7fefb09d5c10>
```



```
corrmat = train.corr(method='pearson')
f, ax = plt.subplots(figsize=(16,16))
sns.heatmap(corrmat,vmax=.8,square=True,cmap='RdYlBu')
plt.show()
```



```
plt.figure(figsize=(12, 12))
k = 10 #Top k variabel yang berkorelasi dengan SalePrice
columns = corrmat.nlargest(k,'SalePrice')['SalePrice'].index
cm = np.corrcoef(train[columns].values.T)
sns.set(font_scale=1.25)
hm = sns.heatmap(cm, cbar=True, annot=True, fmt='.2f', square=True,
annot_kws={'size': 12},
                 yticklabels=columns.values,
xticklabels=columns.values, cmap='RdYlBu')
plt.show()
```



```
train = train[train.GrLivArea < 4500]

total = test.isna().sum().sort_values(ascending=False)
#concatenate missing data into dataframe
missing = pd.concat([total],axis=1, keys=['Total'])
missing.head(45)
```

```
               Total
PoolQC          1456
MiscFeature     1408
Alley           1352
Fence           1169
FireplaceQu      730
LotFrontage      227
GarageYrBlt       78
GarageQual        78
GarageFinish      78
GarageCond        78
GarageType        76
```

```
BsmtCond          45
BsmtQual          44
BsmtExposure      44
BsmtFinType1      42
BsmtFinType2      42
MasVnrType        16
MasVnrArea        15
MSZoning           4
BsmtHalfBath       2
Utilities          2
Functional         2
BsmtFullBath       2
BsmtFinSF1         1
BsmtFinSF2         1
BsmtUnfSF          1
KitchenQual        1
TotalBsmtSF        1
Exterior2nd        1
GarageCars         1
Exterior1st        1
GarageArea         1
SaleType           1
MiscVal            0
BedroomAbvGr       0
KitchenAbvGr       0
YrSold             0
TotRmsAbvGrd       0
MoSold             0
Fireplaces         0
PoolArea           0
HalfBath           0
ScreenPorch        0
3SsnPorch          0
EnclosedPorch      0
```

```
train = train.drop(missing[missing.Total>0].index, axis=1)
```

```
test = test.dropna(axis=1)
test = test.drop(["Electrical"], axis=1)
```

Regression

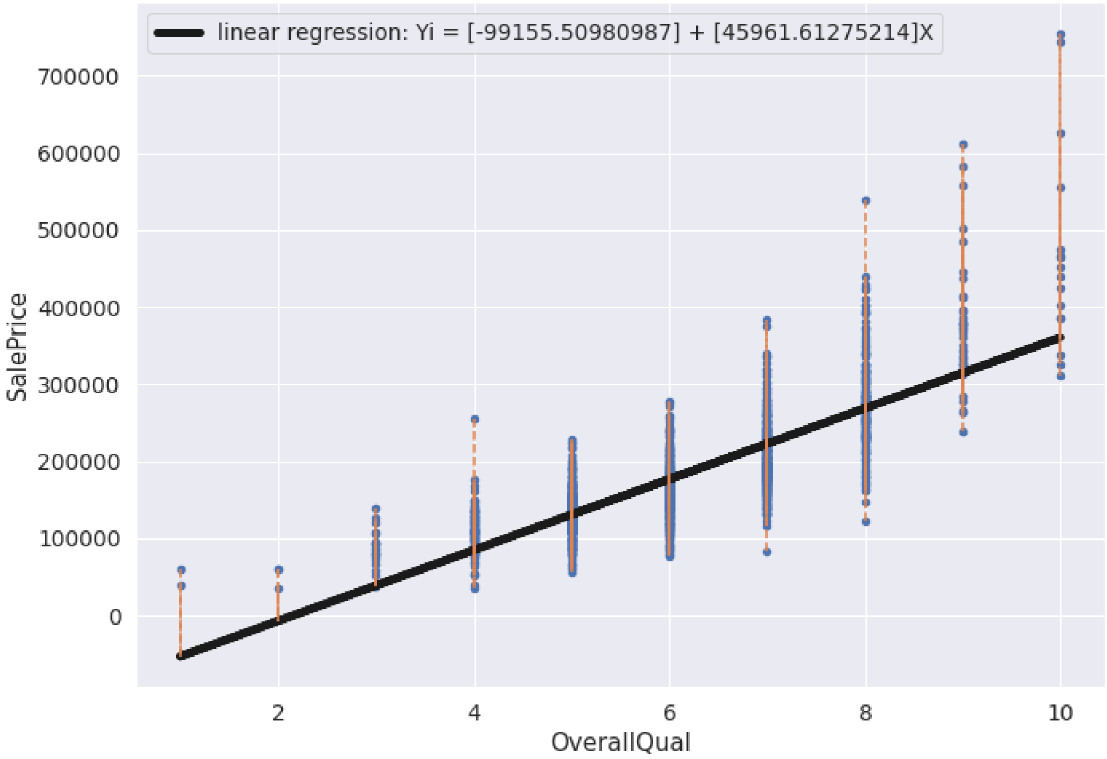
```
predictor = ['OverallQual'] #X
out = ['SalePrice'] #(Yi)

#Rumus LR Yi=bo+b1X
model = LinearRegression()
model.fit(train[predictor], train[out])

print(f'Intercept: {model.intercept_:}') #Nilai untuk b0 atau c
print(f'Coefficient: {model.coef_[0]:}') #Nilai untuk b1 atau m
```

```
fitted = model.predict(train[predictor]) #(Yi hat=Ypred)
residuals = train[out] - fitted #e=Yi-Yhat
```

```
Intercept: [-99155.50980987]
Coefficient: [45961.61275214]
```

```
ax = train.plot.scatter(x='OverallQual', y='SalePrice', figsize=(10,
7))
ax.plot(train.OverallQual, fitted, linewidth=5, color='k',
label=f'linear regression: Yi = {model.intercept_:} +
{model.coef_[0]}X')
for x, yactual, yfitted in zip(train.OverallQual, train.SalePrice,
fitted):
    ax.plot((x, x), (yactual, yfitted), '--', color='C1')
plt.tight_layout()
plt.legend()   #fungsi plt.legend () melacak gaya dan warna garis, dan
mencocokkannya dengan label yang benar.
plt.show()
```

```
*c* argument looks like a single numeric RGB or RGBA sequence, which
should be avoided as value-mapping will have precedence in case its
length matches with *x* & *y*.  Please use the *color* keyword-
argument or provide a 2-D array with a single row if you intend to
specify the same RGB or RGBA value for all points.
/usr/local/lib/python3.7/dist-packages/numpy/core/shape_base.py:65:
VisibleDeprecationWarning: Creating an ndarray from ragged nested
sequences (which is a list-or-tuple of lists-or-tuples-or ndarrays
with different lengths or shapes) is deprecated. If you meant to do
this, you must specify 'dtype=object' when creating the ndarray.
  ary = asanyarray(ary)
```



Prediction

```
full_df = pd.concat([train,test])
```

```
full_df = pd.get_dummies(full_df)
```

```
X = full_df.iloc[train.index]
X_test = full_df.iloc[test.index]
```

```
X = X.drop(['SalePrice'],axis=1)
```

```
X.shape
```

```
(1458, 154)
```

```
y = train.SalePrice
y.shape
```

```
(1458,)
```

```
from sklearn.model_selection import train_test_split
X_train, X_val, y_train, y_val = train_test_split(X, y,
train_size=0.8, random_state=42)
```

```
X.isna().sum().sort_values(ascending=False)
```

```
Id                          0
RoofMatl_CompShg            0
HouseStyle_SLvl             0
```

```
RoofStyle_Flat           0
RoofStyle_Gable          0
RoofStyle_Gambrel        0
RoofStyle_Hip            0
RoofStyle_Mansard        0
RoofStyle_Shed           0
RoofMatl_Membran         0
ExterQual_TA             0
RoofMatl_Metal           0
RoofMatl_Roll            0
RoofMatl_Tar&Grv         0
RoofMatl_WdShake         0
RoofMatl_WdShngl         0
ExterQual_Ex             0
ExterQual_Fa             0
HouseStyle_SFoyer        0
HouseStyle_2Story        0
HouseStyle_2.5Unf        0
HouseStyle_2.5Fin        0
Condition2_Feedr         0
Condition2_Norm          0
Condition2_PosA          0
Condition2_PosN          0
Condition2_RRAe          0
Condition2_RRAn          0
Condition2_RRNn          0
BldgType_1Fam            0
BldgType_2fmCon          0
BldgType_Duplex          0
BldgType_Twnhs           0
BldgType_TwnhsE          0
HouseStyle_1.5Fin        0
HouseStyle_1.5Unf        0
HouseStyle_1Story        0
ExterQual_Gd             0
ExterCond_Ex             0
MSSubClass               0
Electrical_SBrkr         0
HeatingQC_TA             0
CentralAir_N             0
CentralAir_Y             0
Electrical_FuseA         0
Electrical_FuseF         0
Electrical_FuseP         0
Electrical_Mix           0
PavedDrive_N             0
ExterCond_Fa             0
PavedDrive_P             0
PavedDrive_Y             0
SaleCondition_Abnorml    0
```

```
SaleCondition_AdjLand    0
SaleCondition_Alloca     0
SaleCondition_Family     0
SaleCondition_Normal     0
HeatingQC_Po             0
HeatingQC_Gd             0
HeatingQC_Fa             0
HeatingQC_Ex             0
ExterCond_Gd             0
ExterCond_Po             0
ExterCond_TA             0
Foundation_BrkTil        0
Foundation_CBlock        0
Foundation_PConc         0
Foundation_Slab          0
Foundation_Stone         0
Foundation_Wood          0
Heating_Floor            0
Heating_GasA             0
Heating_GasW             0
Heating_Grav             0
Heating_OthW             0
Heating_Wall             0
Condition2_Artery        0
Condition1_RRNn          0
Condition1_RRNe          0
LotShape_IR1             0
ScreenPorch              0
PoolArea                 0
MiscVal                  0
MoSold                   0
YrSold                   0
Street_Grvl              0
Street_Pave              0
LotShape_IR2             0
Condition1_RRAn          0
LotShape_IR3             0
LotShape_Reg             0
LandContour_Bnk          0
LandContour_HLS          0
LandContour_Low          0
LandContour_Lvl          0
LotConfig_Corner         0
3SsnPorch                0
EnclosedPorch            0
OpenPorchSF              0
WoodDeckSF               0
LotArea                  0
OverallQual              0
OverallCond              0
```

```
YearBuilt                0
YearRemodAdd             0
1stFlrSF                 0
2ndFlrSF                 0
LowQualFinSF             0
GrLivArea                0
FullBath                 0
HalfBath                 0
BedroomAbvGr             0
KitchenAbvGr             0
TotRmsAbvGrd             0
Fireplaces               0
LotConfig_CulDSac        0
LotConfig_FR2            0
LotConfig_FR3            0
Neighborhood_NWAmes      0
Neighborhood_NridgHt     0
Neighborhood_OldTown     0
Neighborhood_SWISU       0
Neighborhood_Sawyer      0
Neighborhood_SawyerW     0
Neighborhood_Somerst     0
Neighborhood_StoneBr     0
Neighborhood_Timber      0
Neighborhood_Veenker     0
Condition1_Artery        0
Condition1_Feedr         0
Condition1_Norm          0
Condition1_PosA          0
Condition1_PosN          0
Condition1_RRAe          0
Neighborhood_NoRidge     0
Neighborhood_NPkVill     0
LotConfig_Inside         0
Neighborhood_NAmes       0
LandSlope_Gtl            0
LandSlope_Mod            0
LandSlope_Sev            0
Neighborhood_Blmngtn     0
Neighborhood_Blueste     0
Neighborhood_BrDale      0
Neighborhood_BrkSide     0
Neighborhood_ClearCr     0
Neighborhood_CollgCr     0
Neighborhood_Crawfor     0
Neighborhood_Edwards     0
Neighborhood_Gilbert     0
Neighborhood_IDOTRR      0
Neighborhood_MeadowV     0
Neighborhood_Mitchel     0
```

```
SaleCondition_Partial    0
dtype: int64
```

```
from sklearn.linear_model import LinearRegression
from scipy.stats import zscore
regressor = LinearRegression()
regressor.fit(X_train, y_train)
regressor.score(X_val, y_val)
```

```
0.014656716430643923
```

```
X_test = X_test.drop(["SalePrice"], axis=1)
```

```
y_preds = regressor.predict(X_test)
```