**NAME OF PROJECT**

**HOUSING PRICE PREDICTION**

**SUBMITTED BY**

**PANKAJ ADHIKARI**

****

1. **INTRODUCTION**

Owning a House has always been a dream for people who have lived and work in big cities. With the explosion of urban population, demand for Housing has been steadily rising for the past few decades creating a void that needs to be fill. This in turn has provided business opportunity to the real estate developers for building houses and apartment that can meet the supply and demand. In order to attract the customers with various types of housing, developers needs to have the knowhow of housing market, their customers, location, cost of construction and so on. One of such Housing Company named Surprise Housing, which has their presence in US, have decided to bring their expertise to Australia and help in the growing demand. This will in turn also help them in expansion of the business in Australia. To do so they need the help of Data Analysis which will help them buy the house in prices below market value and sell them at higher prices. Hence, they have collected excessive data about Australian Housing market that can help them buy prospective properties and to make a decision on their purchase and sale value.

1. **PROBLEM DEFINITION**

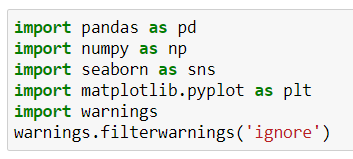
The Dataset provided different parameters that determine the price of the House. These parameters can be used to build a Machine Learning Model to predict the Sale Price of the House giving the company to decide whether to buy the property or not. This Dataset provide the challenge of analysing and quantifying the parametrs that contribute directly to the Sale Price of the house.

1. **DATA ANALYSIS**

In this section we perform Exploratory Data Analysis (EDA) so as to gain insight on Data by modification, Visualisation.

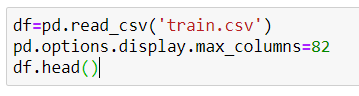
**IMPORTING LIBRARIES**

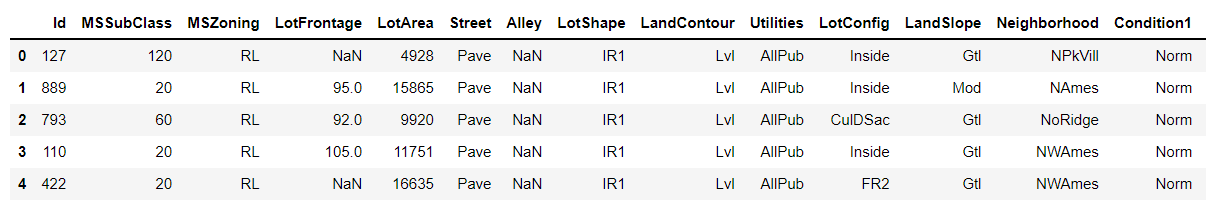
We begin by importing important Libraries



**UPLOADING DATA**

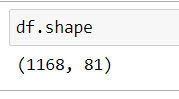
Uploading the data set into a Data Frame and printing the first five rows





**SHAPE**

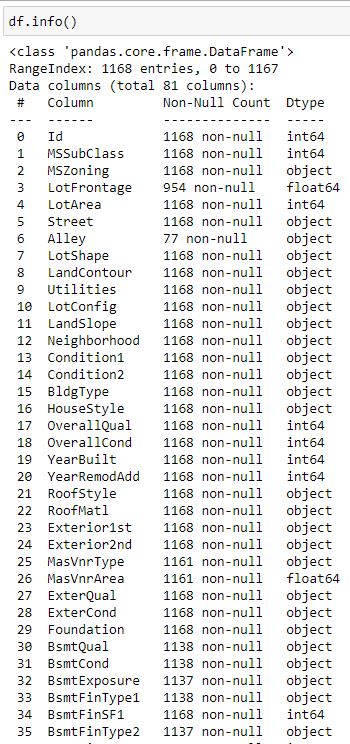
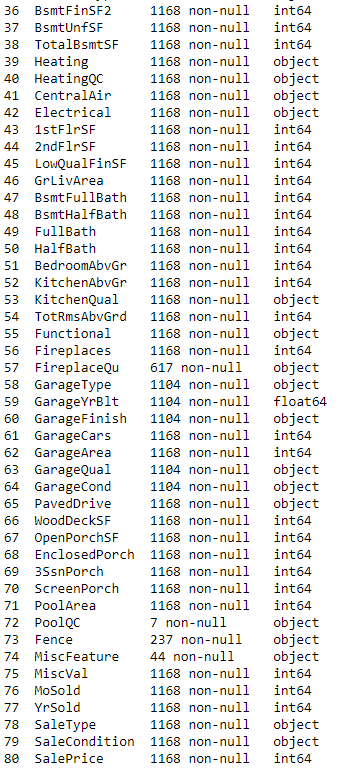
This show that Data has 1168-rows and 81-columns



**INFO**

This shows the datatype along with number of data available.

*Output:*

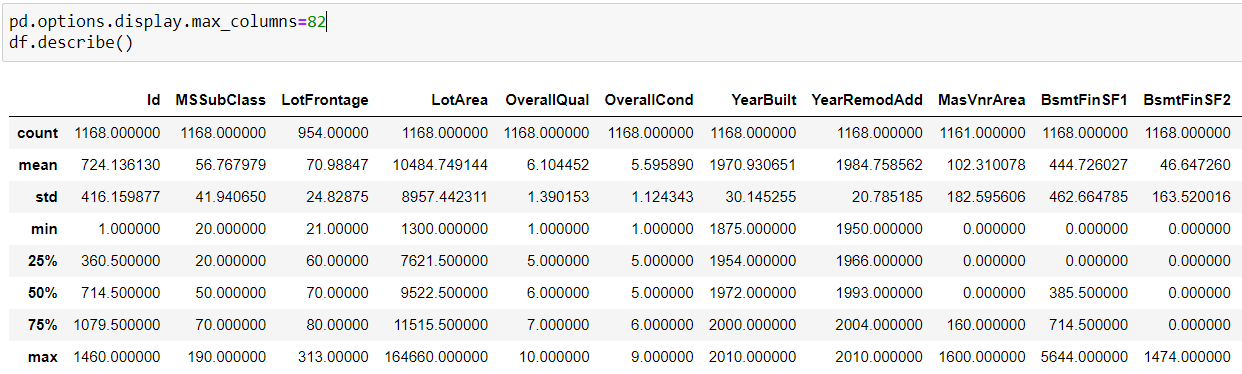
Observation:

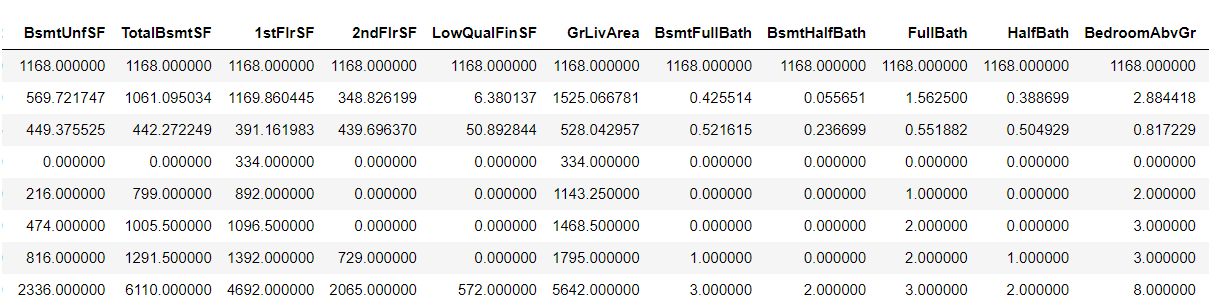
* Columns like Alley, PoolQc, Fence, MiscFeature, FireplaceQu have very less data hence these columns has to be dropped.
* Columns like MasVnrType, MasVnrArea, BsmtQual, BsmtCond, BsmtExposure, BsmtFinType1, BsmtFinType2, GarageType, GarageYrBlt, GarageFinish, GarageQual, GarageCond, GarageFinish have missing data which needs to be filled with mode of data.

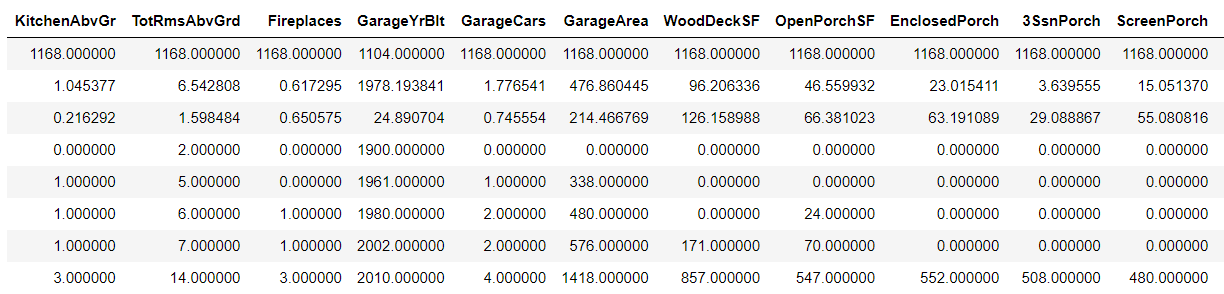
**DATA DESCRIBE**

Describing the important parameters of the data.

*Output:*









Observations:

1. Columns like OverallQual, OverallCond has very low standard deviation signifying data to be clustered around mean value.
2. Column BsmtFinSF2 has high standard deviation signifying data to be highly spread

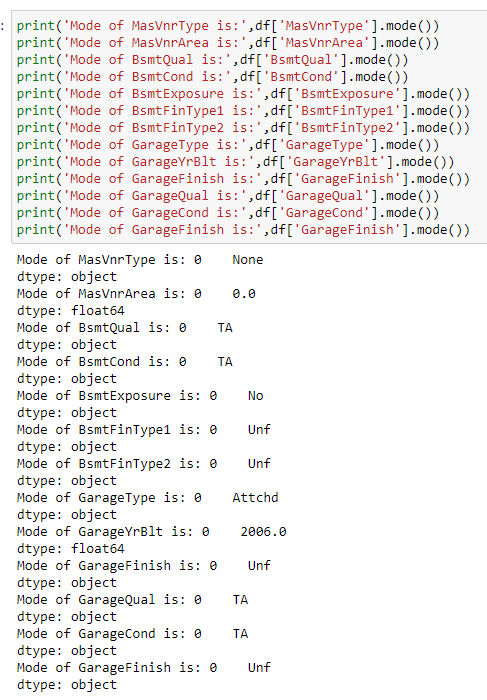
**DROPPING OF COLUMNS**

* Columns like Alley, PoolQc, Fence, MiscFeature, FireplaceQu have very less data hence these columns are dropped as shown below.
* Id Column has been drop as the value is all unique and does not provide correct result in prediction.

****

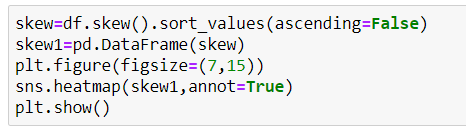
**MISSING DATA**

We Fill the missing data with mode of the data as shown below.

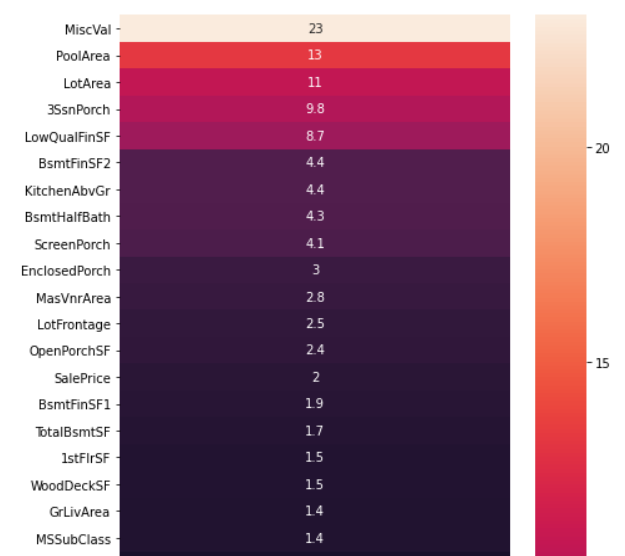
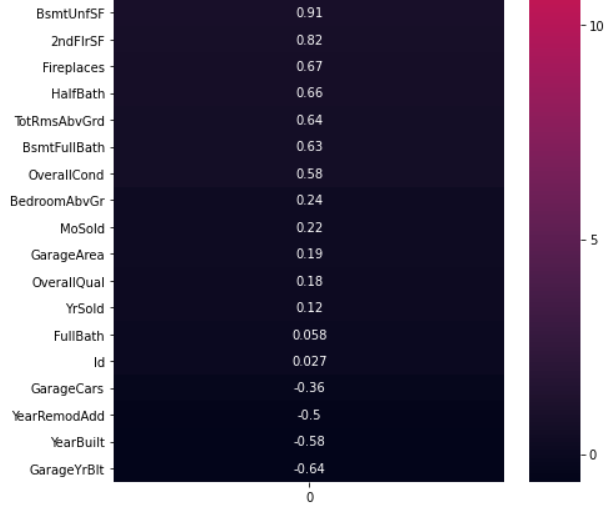
** **

**SKEWNESS OF COLUMNS**

Calculating skewness of Columns and plotting in a Descending Heatmap.



*Output:*

**COORELATION HEAT MAP**

Due to high number of columns we divide the heat map into four group for better visualization:

df.iloc[:41,:41], df.iloc[:41,41:], df.iloc[41:,:41], df.iloc[41:,41:]

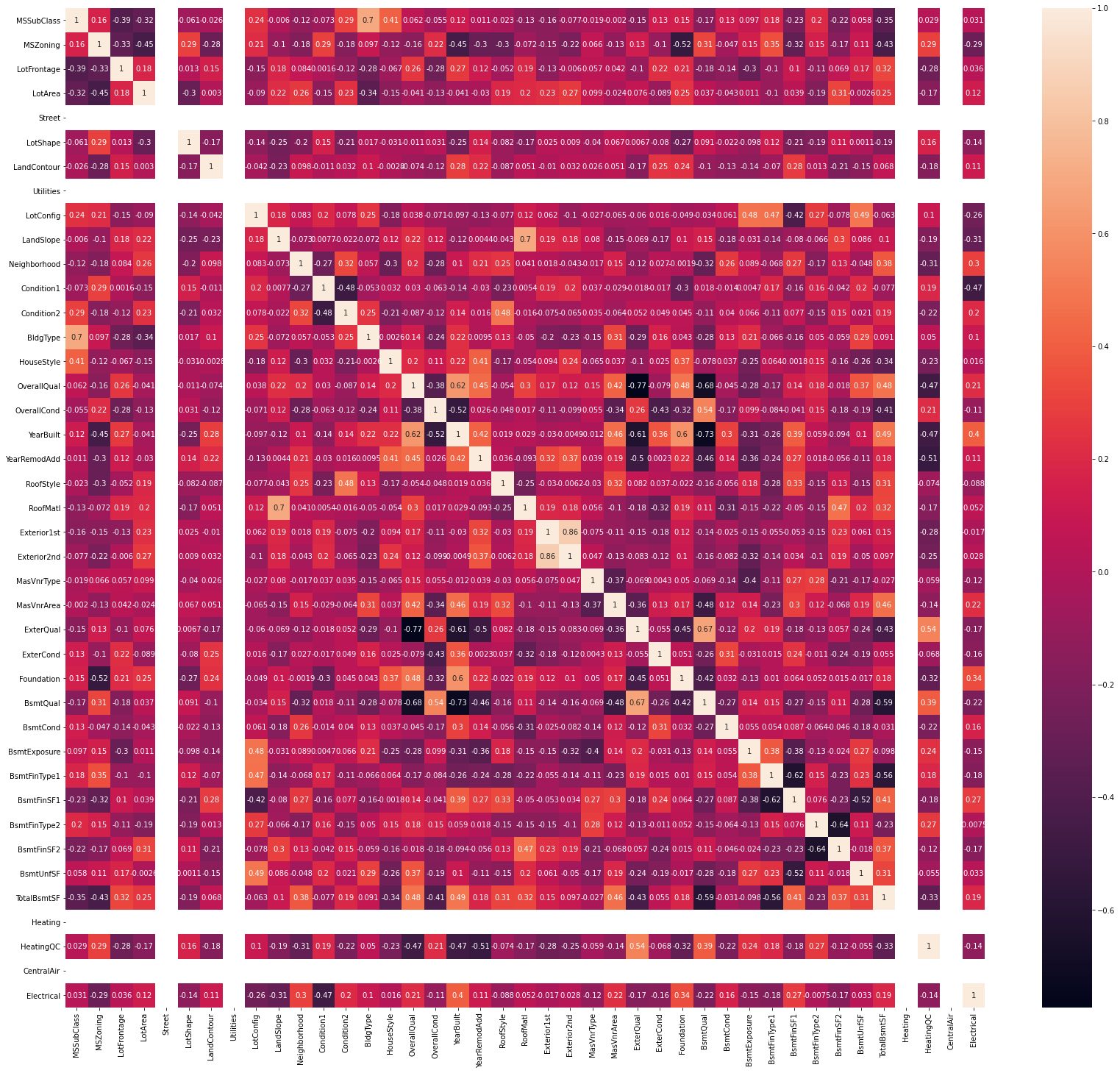
****

Figure : Heatmap-1

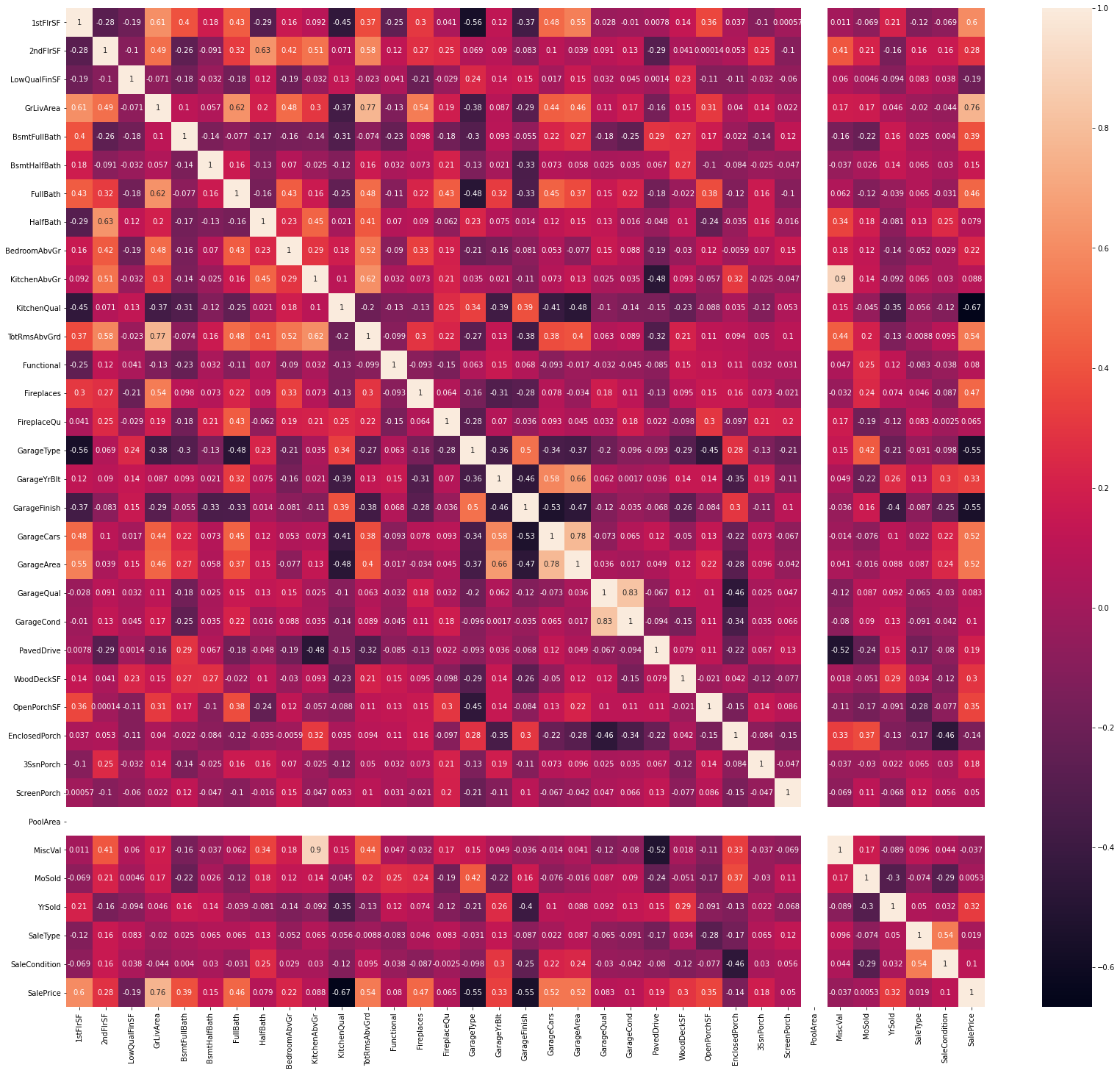
****

Figure : Heatmap-2

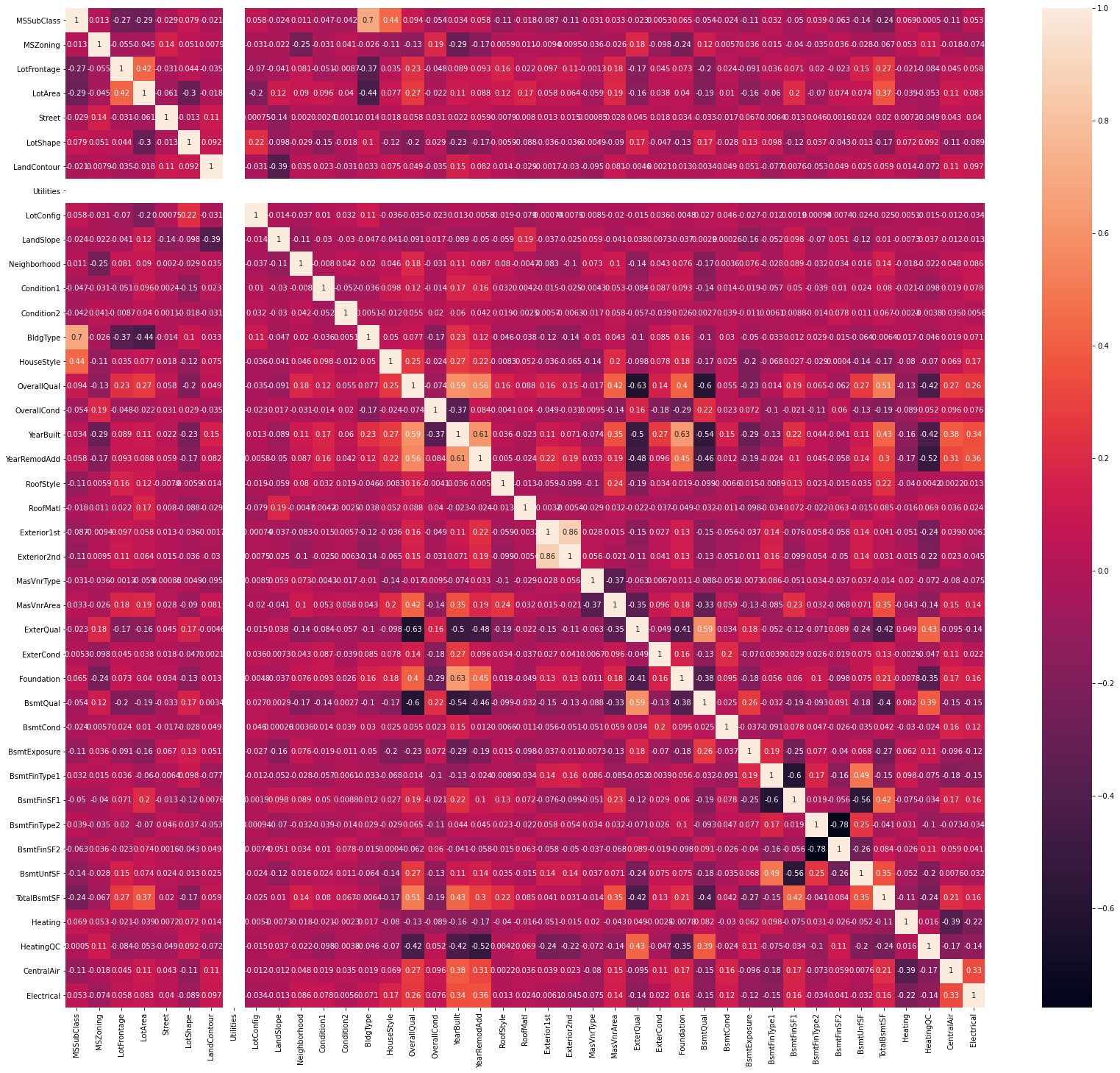
****

Figure : Heatmap-3

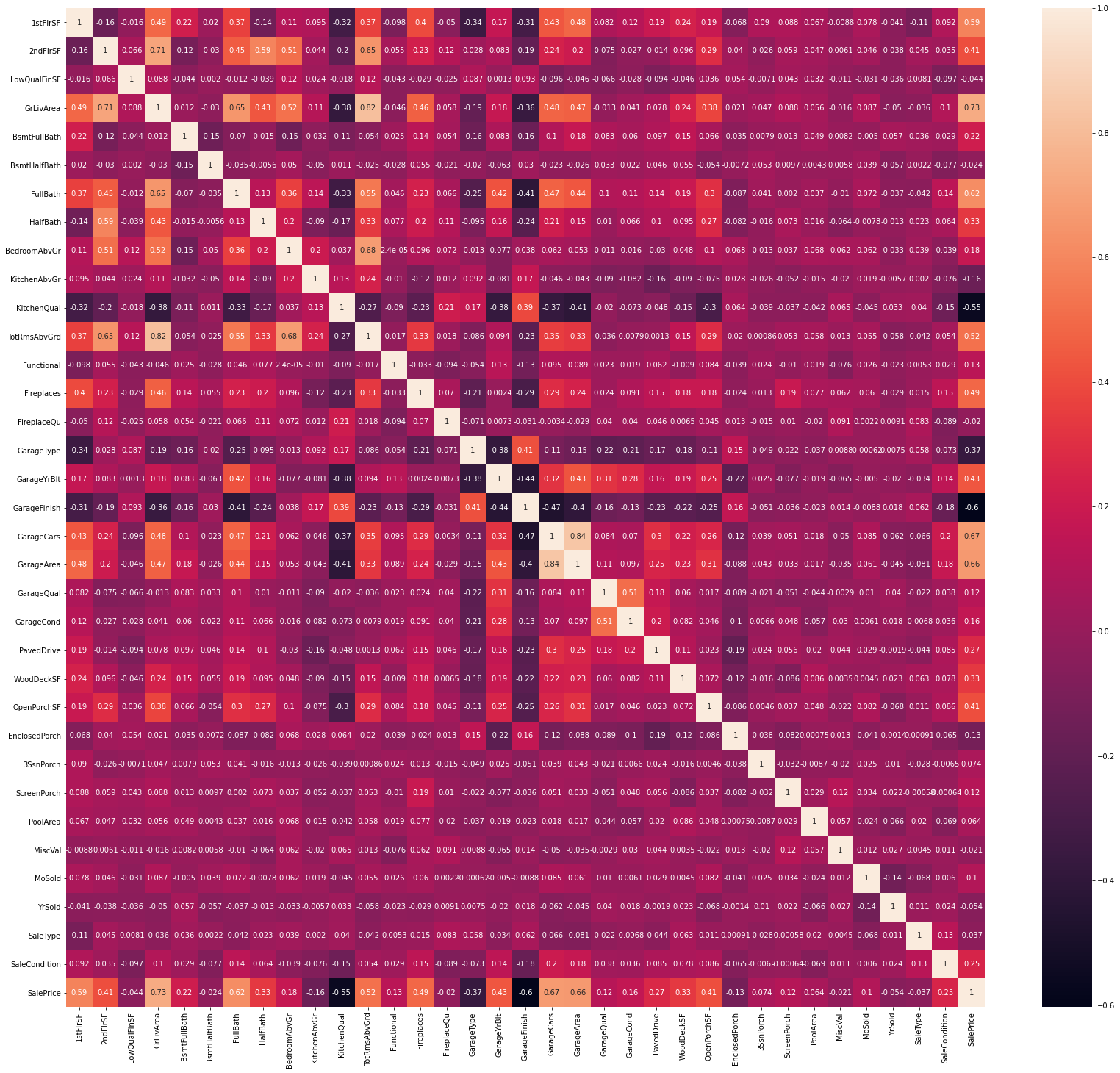
****

Figure : Heatmap-4

Observation:

* In Heatmap-2 Columns KitchenAbvGr and MiscVal has high Correlation of 0.9
* In Heatmap-4 Columns has high Correlation of 0.82
* Columns SalePrice and GarageFinish has low Correlation of -0.6
* In Heatmap-3 Columns BsmtFinSF2 and BsmtFinType2 has low Correlation of -0.78

**PLOTS AND GRAPHS**

**PLOT-1**

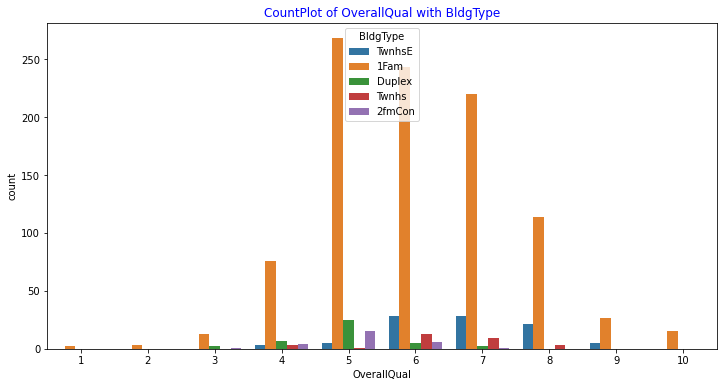
****

Figure : CountPlot of OverallQual with BldgType

OBSERVATION:

* 1Fam BldgType have high count
* OverallQual of 5 have the highest Count
* OverallQual of 1,2 and 10 have only 1Fam BldgType

**PLOT-2**

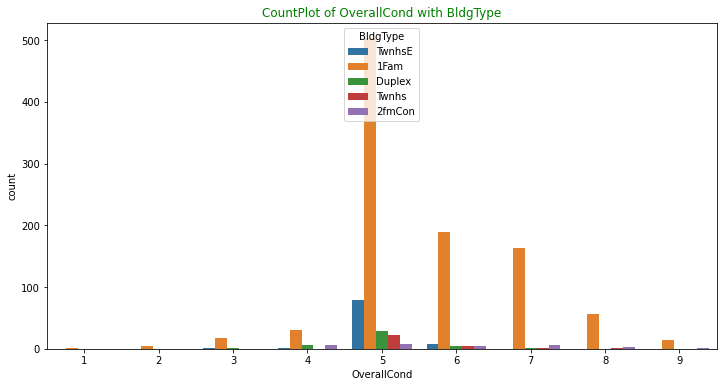
****

Figure : CountPlot of OverallCond with BldgType

OBSERVATIONS:

* OverallCond of 5 have the highest count
* 1Fam BldgType have the highest count
* OverallCond of 10 have the least count

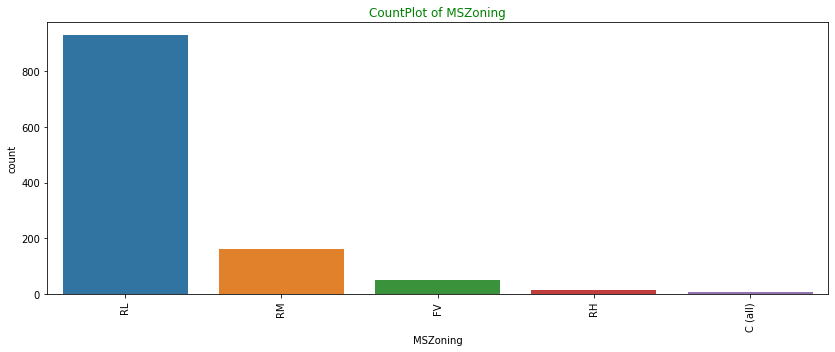
**PLOT-3**

Figure : CountPlot of MSZoning

OBSERVATIONS:

* RL MSZoning have the highest count
* C type MSZoning have the lowest count.

**PLOT-4**

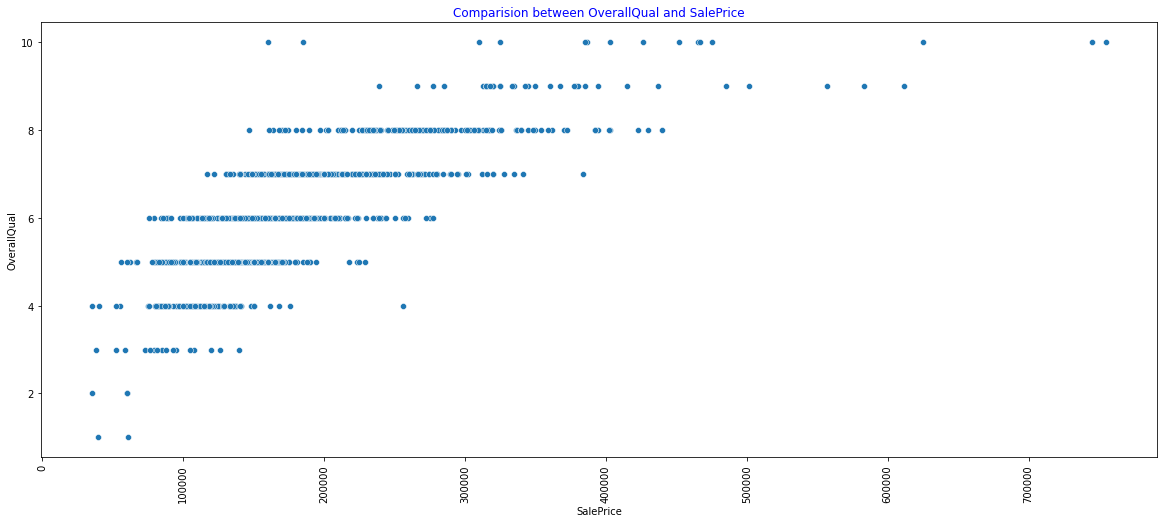
****

Figure : Comparision between OverallQual and SalePrice

OBSERVATION:

* Highest SalePrice is that of OverallQual 10.
* Lowest SalePrice is that of OverallQual 2.
* OverallQual of 1,2,and 3 is of low SalePrice
* OverallQual 10 has range from low to High SalePrice.

**PLOT-5**

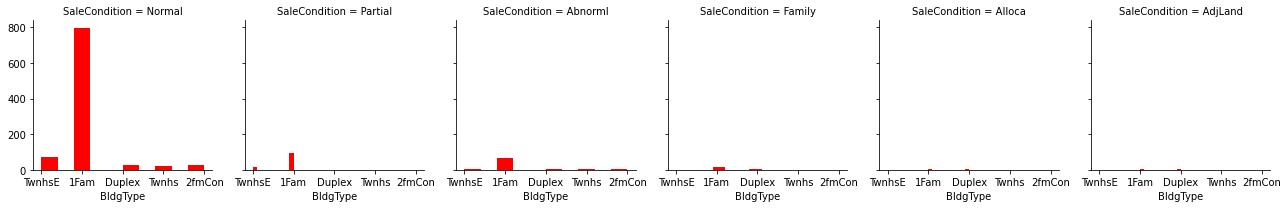
****

Figure : FacetGrid Plot of SaleCondition with Bldg Type

OBSERVATIONS:

* SaleCondition -Normal has the highest count

**PLOT-6**

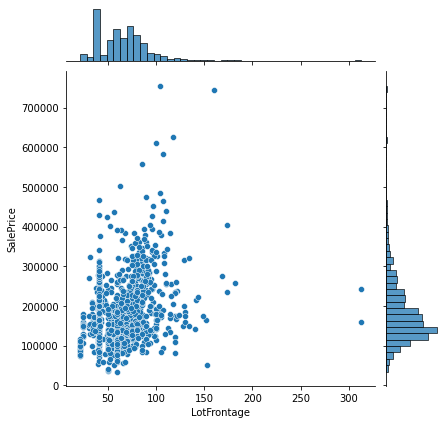
****

Figure : Joint Plot of LotFrontage and SalePrice

OBSERVATION:

* This Joint Plot shows the data is highly spread.
* LotFrontage is concentrated between 20 and 100

**PLOT-7**

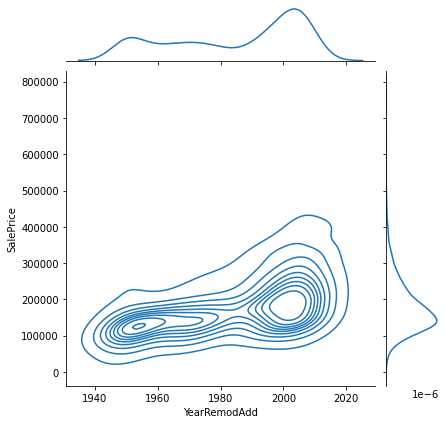
****

Figure : Joint Plot between YearRemodAdd and SalePrice

OBSERVATIONS:

* Data between YearRemodAdd and SalePrice is highly spread
* SalePrice is concentrated between 50k and 300k.

**PLOT-8**

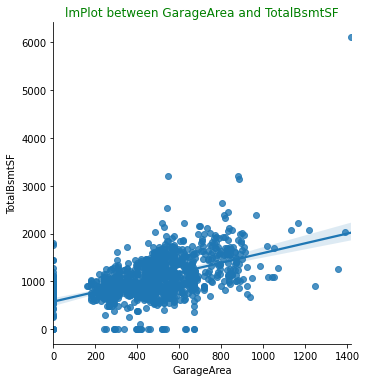
****

Figure : lmPlot Between GarageArea and TotalBsmtSF

OBSERVATIONS:

* High density of data between 200 and 900 Garage area.
* We have an outlier at 6000 TotalBsmtSF mark.
* Few of the data have zero TotalBsmtSF

**PLOT-9**

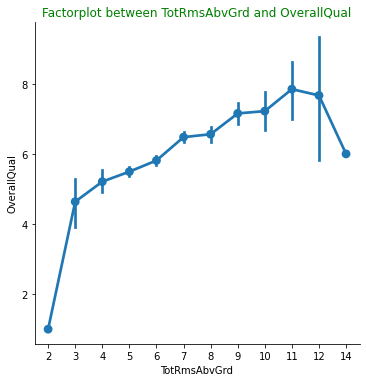
****

Figure : FactorPlot Between TotRmsAbvGrd and OverallQual

OBSERVATION

* There is steep rise in OverallQual between 2-3 TotRmsAbvGrd.
* TotRmsAbvGrd rises till 11 after that there is decline

**PLOT-10**

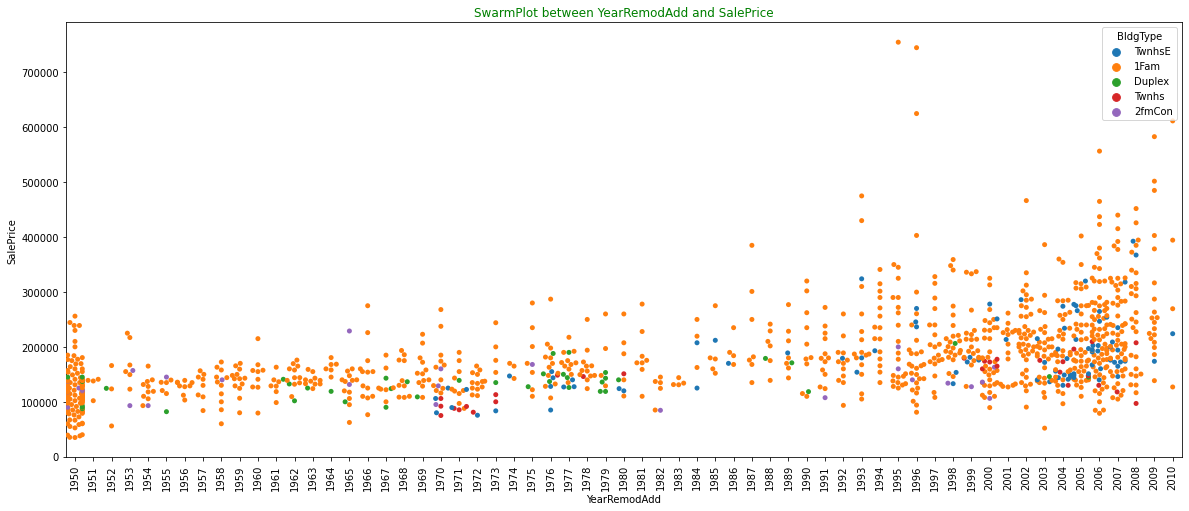
****

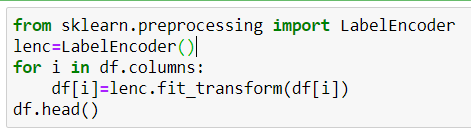
Figure : SwarmPlot between YearRemodAdd and SalePrice

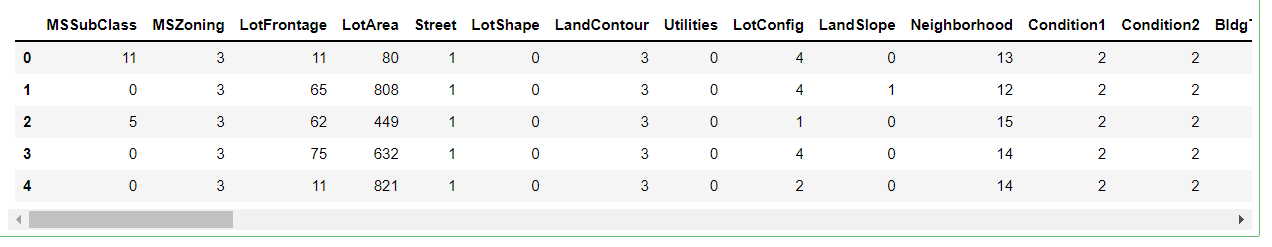
OBSERVATION

* YearRemodAdd of 1950 has low SalePrice.
* YearRemodAdd between 2000 and 2008 has high data density.
* Twnhs BldgType has high YearRemodAdd between 1970 and 1973
* Prior to 1993 SalePrice is low.
* Number of High SalePrice is quite low.

**LABEL ENCODING OF DATASET**

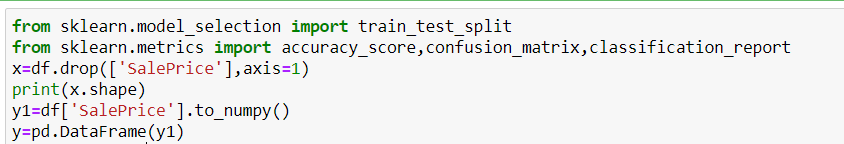
Label Encoding is performed as the data have different datatype, this encoding will later be used in Machine Learning as shown below.



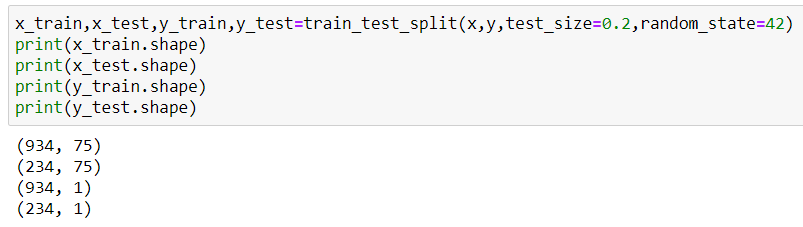
*Output*

**TRAIN TEST SPLIT**

We split the dataset into training and testing (x and y respectively) so that it can be used in Machine Learning for prediction of the Target variable as shown below. The value of x train, x test, y train, y test is shown below.



*Output*

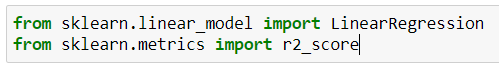


**MACHINE LEARNING MODEL:**

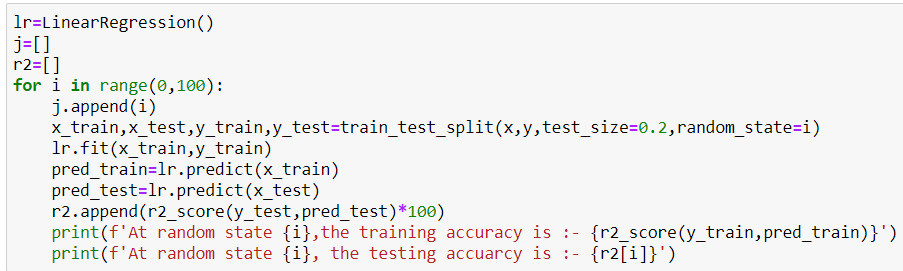
We have seen that the target variable (SalePrice) is s continuous value. Hence to predict the continuous value we use Regression Model. The model used is as shown below.

**LINEAR REGRESSION:**

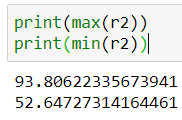
We import the library for Linear Regression and r2 score to determine the accuracy of the model.

****

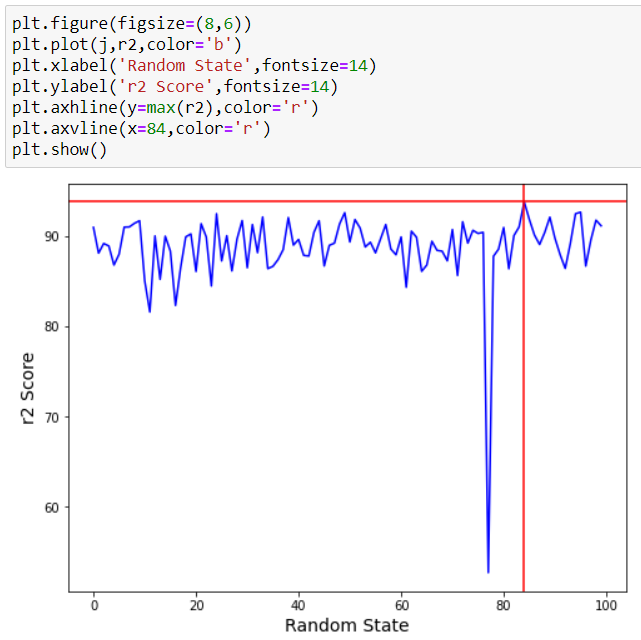
Here we determine the random state where we get the highest accuracy score.

****

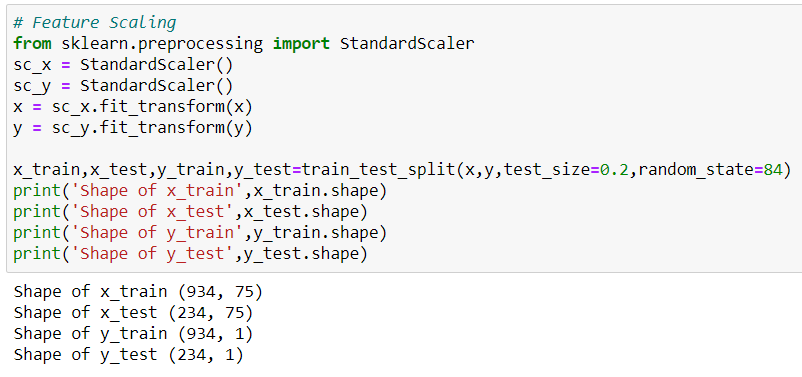
The below suggest that the random state of 84 gives the highest accuracy score of 93.806%, also shown in the graph below.

****

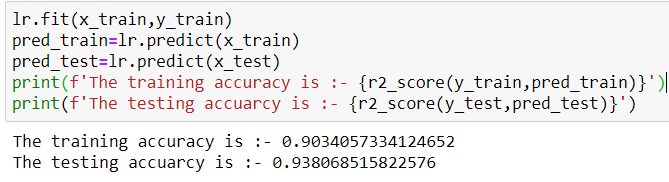
Below Graph showing r2 score with Random State

****

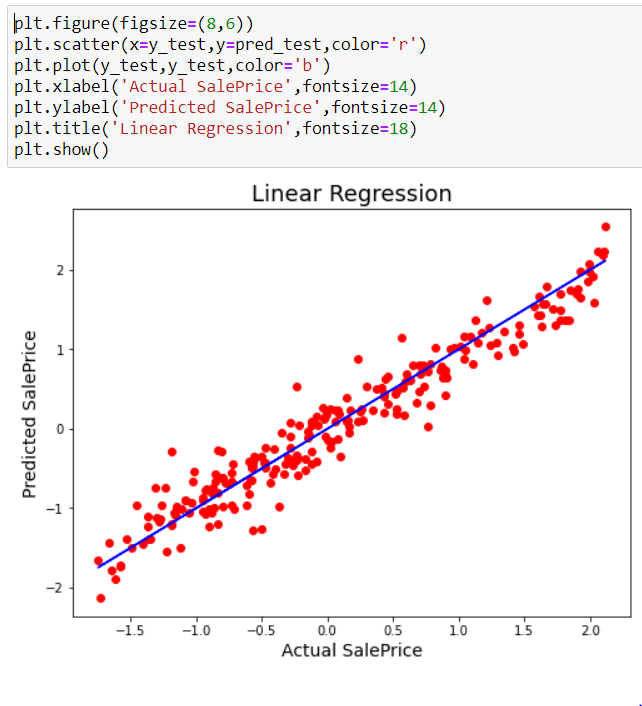
We split the data into Train and Test using after Standard Scaling of x an y

****

We calculate the Accuracy of the Model which comes to 93.80%

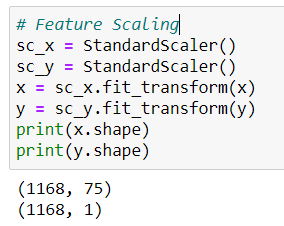
****

This Graph shows the Predicted SalePrice and Actual SalePrice

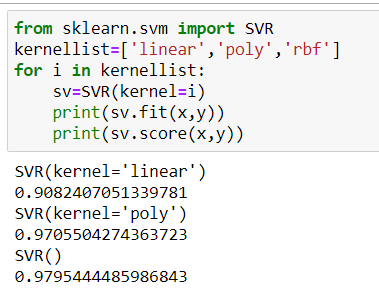
****

**SUPPORT VECTOR REGRESSION:**

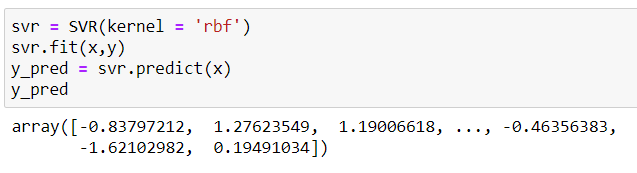
**In this model we first use scaling for x and y variables**

****

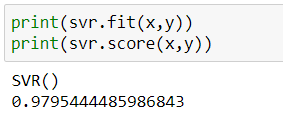
**Finding the best kernel for the model.**

****

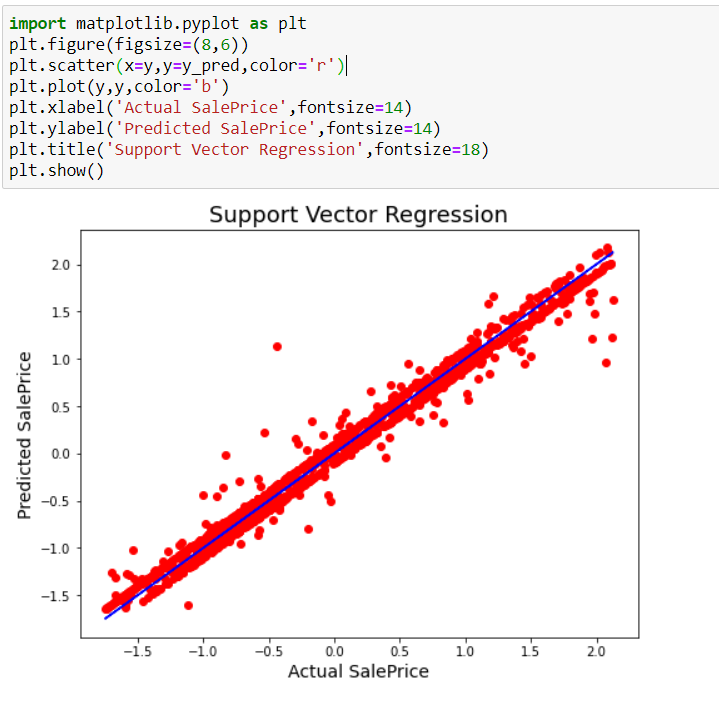
**Using rbf as kernel for prediction of x**

****

**Accuracy score of the model**

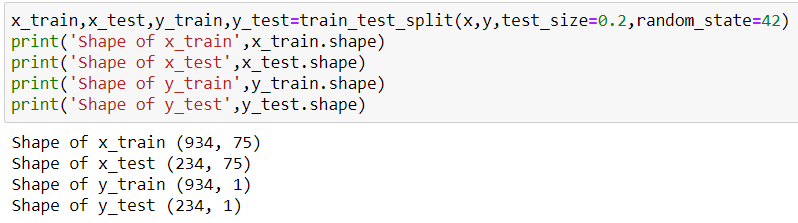
****

**Graphs showing Predicted SalePrice and Actual SalePrice**

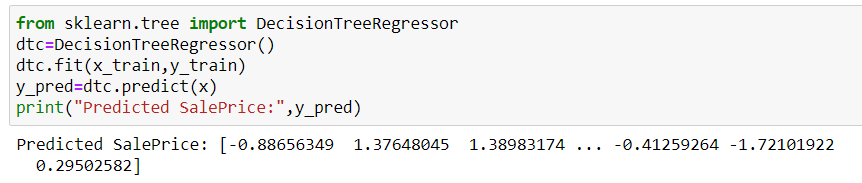
****

**DECISION TREE REGRESSION:**

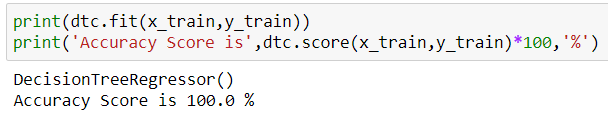
**Splitting the Data for Testing ang Training**

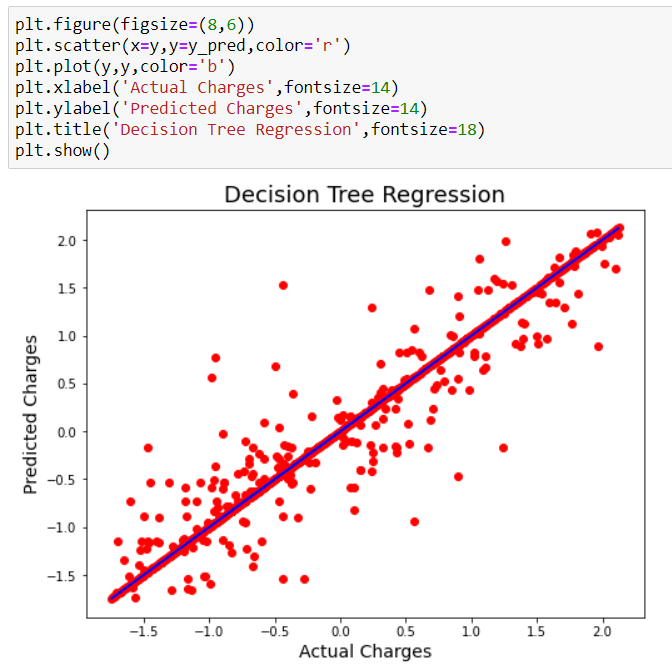
****

**Calculating the Predicted SalePrice**

****

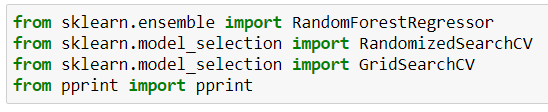
**Accuracy of the Model**

****

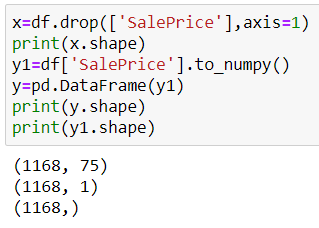
****

**RANDOM FOREST REGRESSION:**

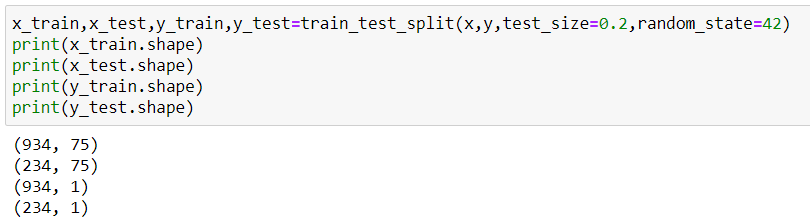
**Importing libraries for Random Forest regressor, Grid searchCV, Randomized Search CV.**

****

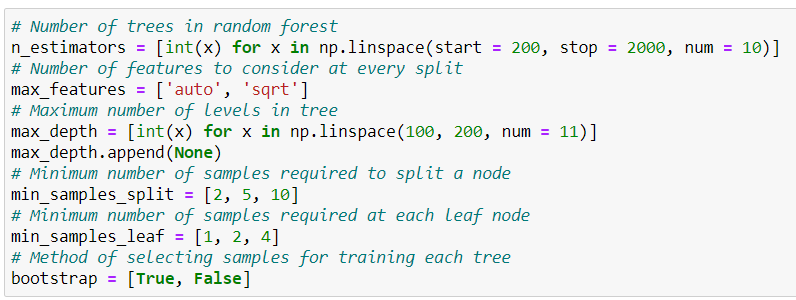
**Deriving feature and label for the data**

****

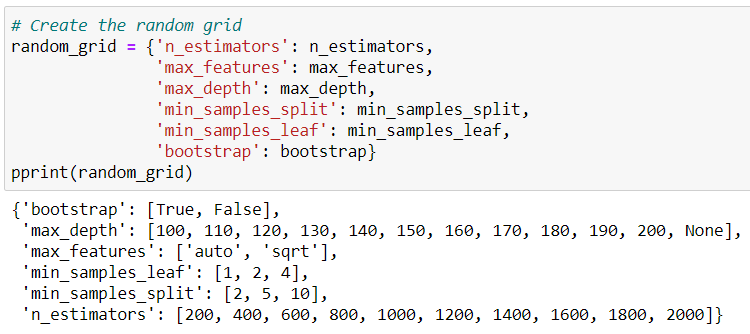
**Spliting data into training and testing**

****

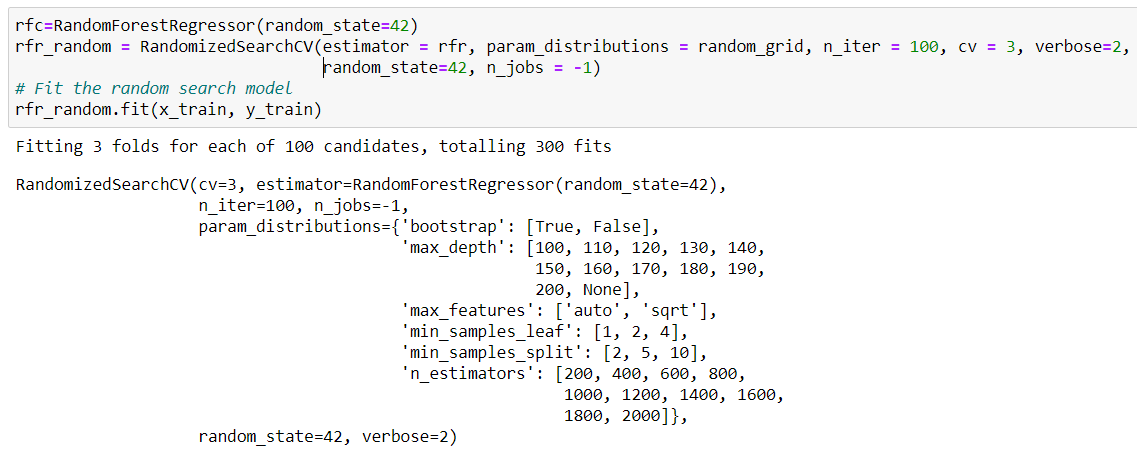
**Parameter for Model**

****

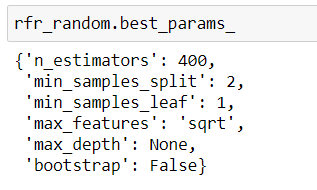
**Creating Random Grid for the Model**

****

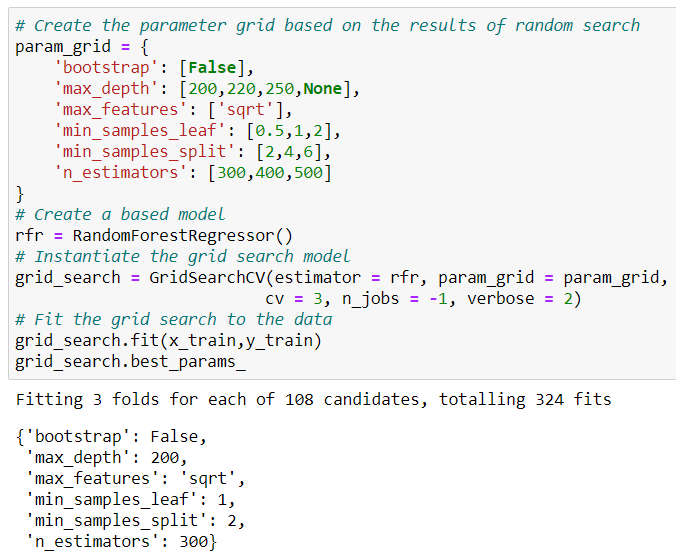
**Fit of Random Search Model**

****

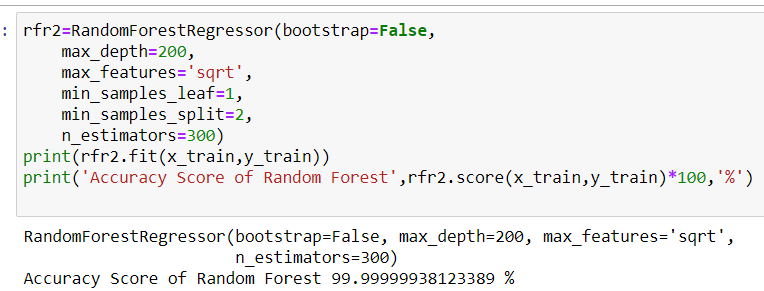
**Best Parameter for the Model**

****

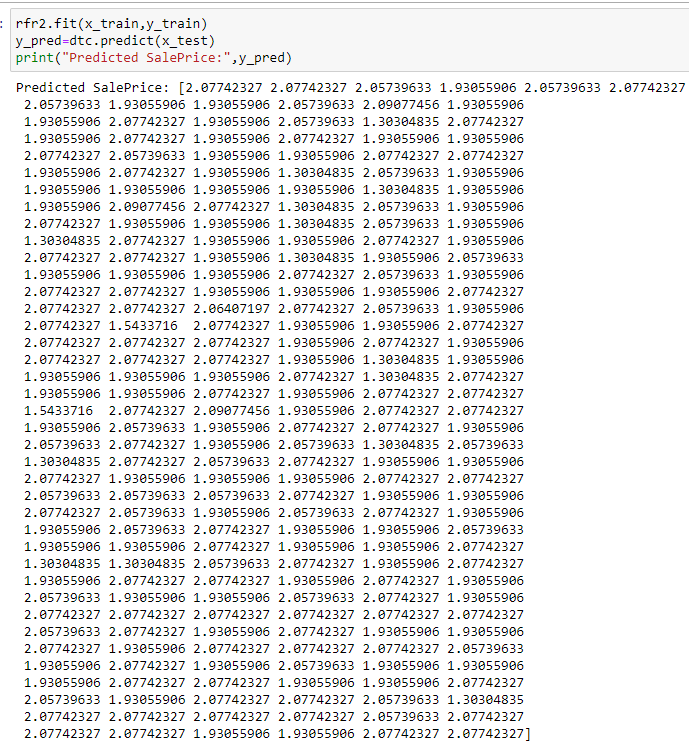
**Creating The Parameter Grid Based on the Result of Random Search**

****

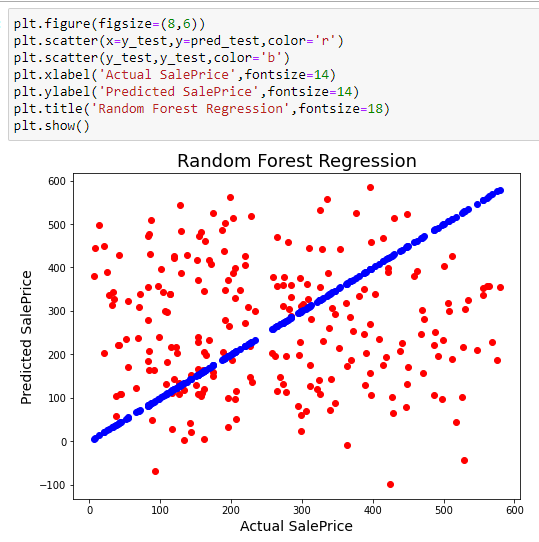
**Accuracy Score of the Model is 99.99%**

****

**Calculating the Predicted SalePrice**

****

**Graph of Predicted and Actual SalePrice**

****

1. **CONCLUSION**

After analysing the result we found the following result.

* The Linear Regression Model gives us an accuracy of **93.80%**
* The Support Vector Regression after best fit gives us an accuracy of **97.95%.**
* The Decision Tree Regression Model gives an Overfitting result as every iteration of parameters provides us only **100%** accuracy.
* Random Forest Regression Model gives us varying accuracy as per the parameters and Grid Search CV. After applying best fit parameter for prediction, it gives us the accuracy of **99.99%.**

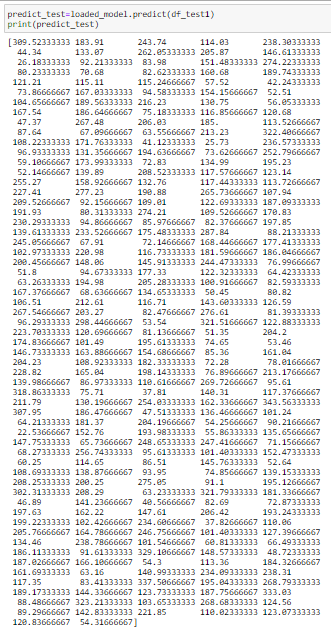
Hence **Random Forest Regression Model** is used for prediction of test dataset**.**

1. **SAVING MODELS**

**We are saving the model in pickle file and load the test dataset as shown below.**

****

**We predict the test dataset as below.**

****

**------ THE END ------**