HOUSING PROJECT

PROBLEM STATEMENT:

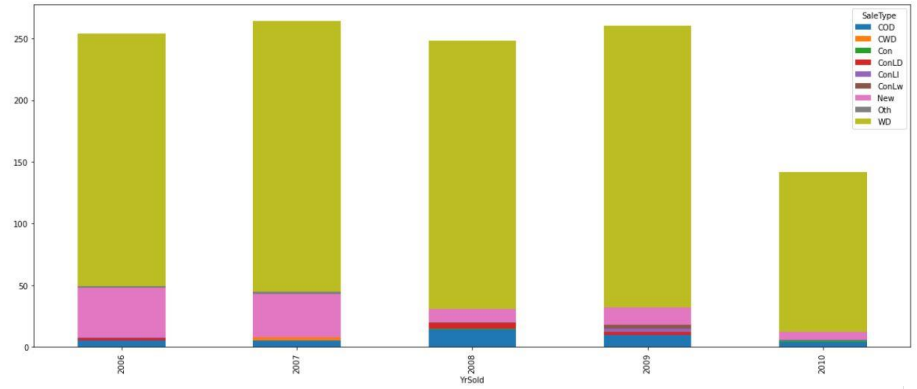
Houses are one of the necessary need of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world's economy. It is a very large market and there are various companies working in the domain. Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases. Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. Our problem is related to one such housing company. A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. The data is provided in the CSV file below. The company is looking at prospective properties to buy houses to enter the market. You are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know: • Which variables are important to predict the price of variable? • How do these variables describe the price of the house?

UNDERSTANDING:

In making of the project, there I realized that the nowadays to have the houses are very necessary need of each and every person around the globe. And those who are not have the houses it become the housing problem to the real world. By doing some workout in the project there we can help others by model training and predicting things makes the previous background to work efficiently.

EDA STEPS AND VISUALIZATIONS

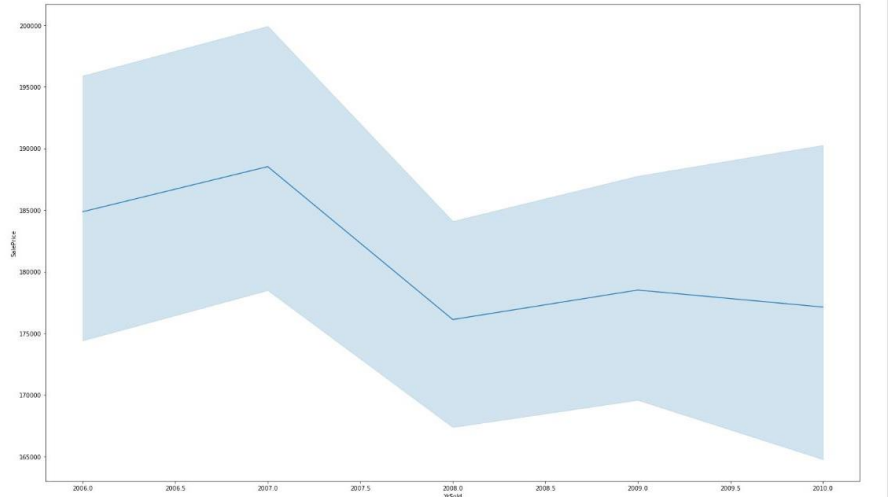
```
In [53]:
1 data=pd.crosstab(df_train['YrSold'], df_train['SaleType'])
2 data.plot.bar(stacked=True,figsize=(20,8))
3 plt.xticks(rotation=90)
4 plt.show()
```



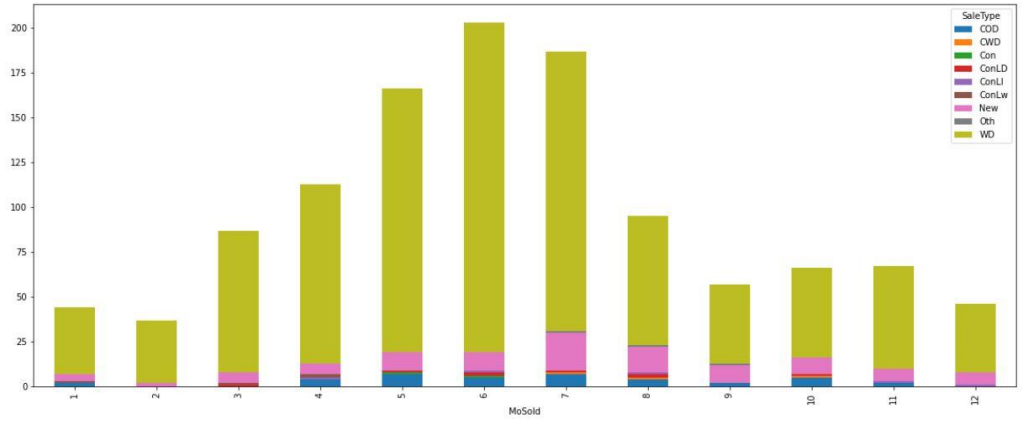
LINEPLOT

```
In [ ]:
1 plt.figure(figsize=(25,15))
2 sns.lineplot(x='YrSold',y='SalePrice',data=df_train)
```

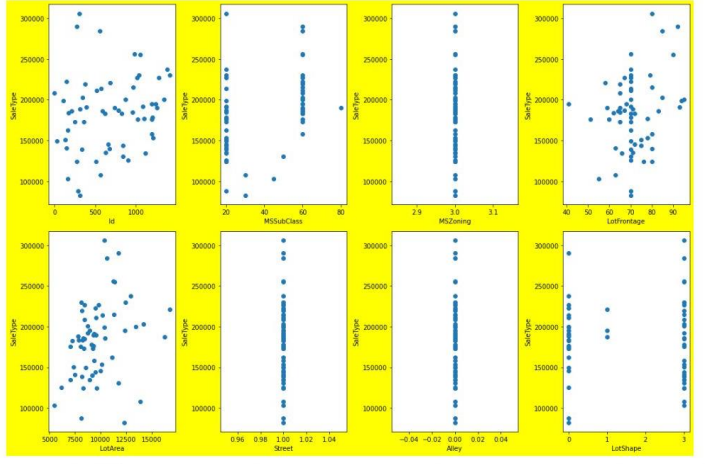
```
Out[ ]: <AxesSubplot:xlabel='YrSold', ylabel='SalePrice'>
```



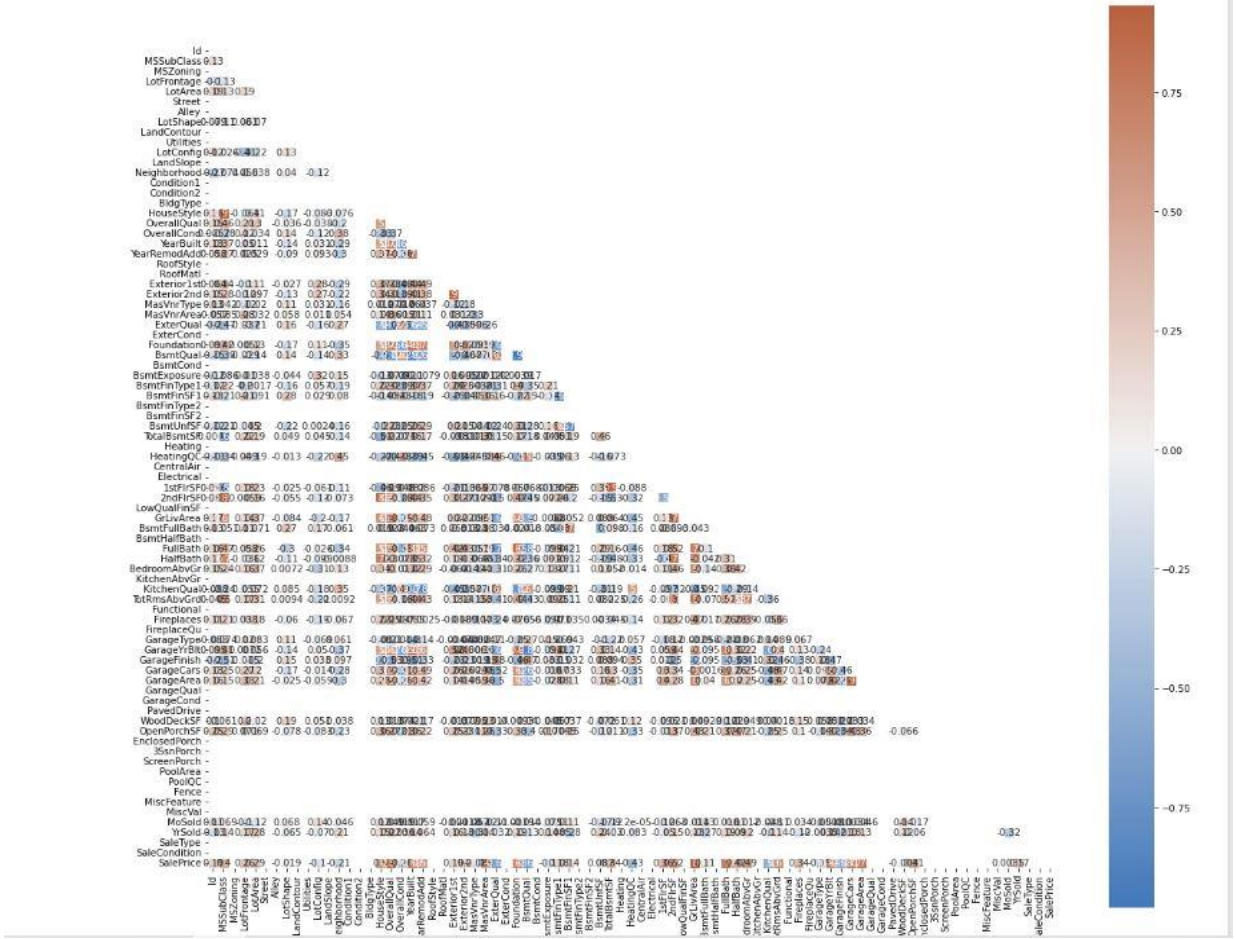

```
In [54]:
1 data=pd.crosstab(df_train['MoSold'], df_train['SaleType'])
2 data.plot.bar(stacked=True,figsize=(20,8))
3 plt.xticks(rotation=90)
4 plt.show()
```




```
In [ ]:
1  # Visualizing relationship
2  plt.figure(figsize=(15,10), facecolor= 'yellow')
3  plotnumber  = 1
4
5  for column in X:
6      if plotnumber<=8  :
7          ax  = plt.subplot(2,4,plotnumber)
8          plt.scatter(X[column],y)
9          plt.xlabel(column,fontsize=10)
10         plt.ylabel('SaleType',fontsize=10)
11     plotnumber+=1
12 plt.tight_layout()
```

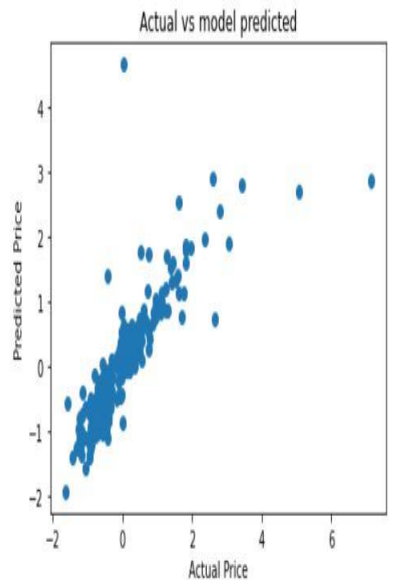


In [91]:
```
corr = MiceImputed.corr()
mask = np.triu(np.ones_like(corr, dtype=np.bool))
f, ax = plt.subplots(figsize=(20,20))
cmap = sns.diverging_palette(250, 25, as_cmap=True)
sns.heatmap(corr, mask=mask, cmap=cmap, vmax=None, center=0,square=True, annot=True, linewidths=.5, cbar_kws={"shrink": .9})
plt.show()
```



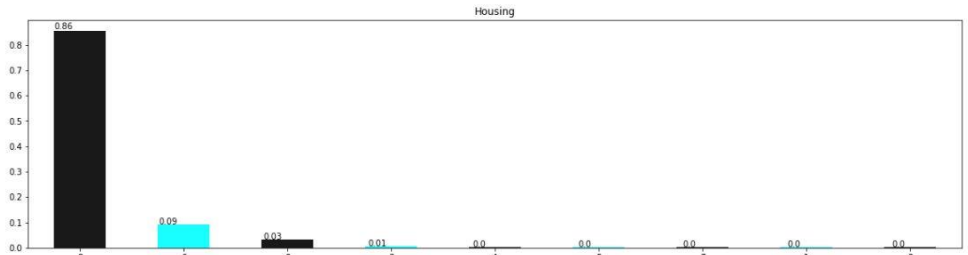
In [106]:
```
plt.scatter(y_test,y_pred)
plt.xlabel('Actual Price')
plt.ylabel('Predicted Price')
plt.title('Actual vs model predicted')
plt.show
```

Out[106]: <function matplotlib.pyplot.show(close=None, block=None)>



In [81]:
```
import matplotlib.pyplot as plt
plt.figure(figsize=(20,5))
ax=df_train.SaleType.value_counts(normalize=True).plot(kind='bar', color=['black', 'cyan'], alpha=0.9, rot=0)
plt.title('Housing')
for i in ax.patches:
    ax.annotate(str(round(i.get_height(),2)),(i.get_x() * 1.01, i.get_height() * 1.01))

plt.show()
```



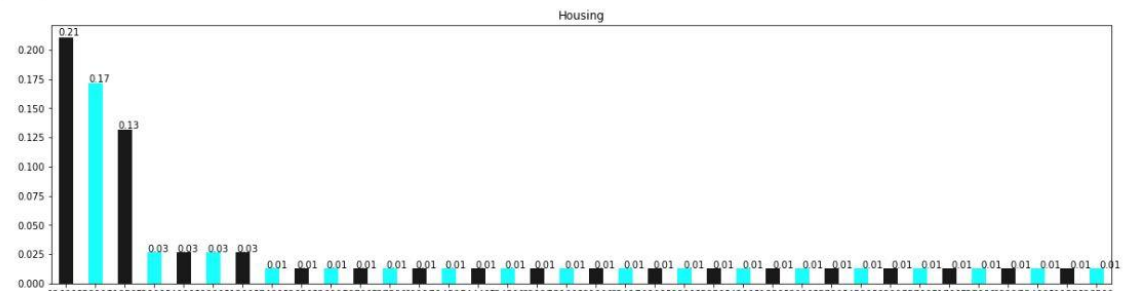
]:
```
from sklearn.utils import resample

no=df_train[df_train.SaleType == 0]
yes=df_train[df_train.SaleType == 1]
yes_oversampled=resample(yes, replace=True, n_samples=len(no), random_state=42)
oversampled = pd.concat([no, yes_oversampled])

plt.figure(figsize=(20,5))

ax=oversampled.SalePrice.value_counts(normalize=True).plot(kind='bar', color=['black', 'cyan'], alpha=0.9, rot=0)
plt.title('Housing')
for i in ax.patches:
    ax.annotate(str(round(i.get_height(),2)),(i.get_x() * 1.01, i.get_height() * 1.01))

plt.show()
```



STEPS AND ASSUMPTIONS TO COMPLETE THE PROJECT:

1. The model of the house price with the available independent variables and exactly the prices are vary with the variables.

2. It can accordingly manipulating the strategy of the areas that will yield high returns as it make easier for the firm whoever take this opportunity in the housing project.

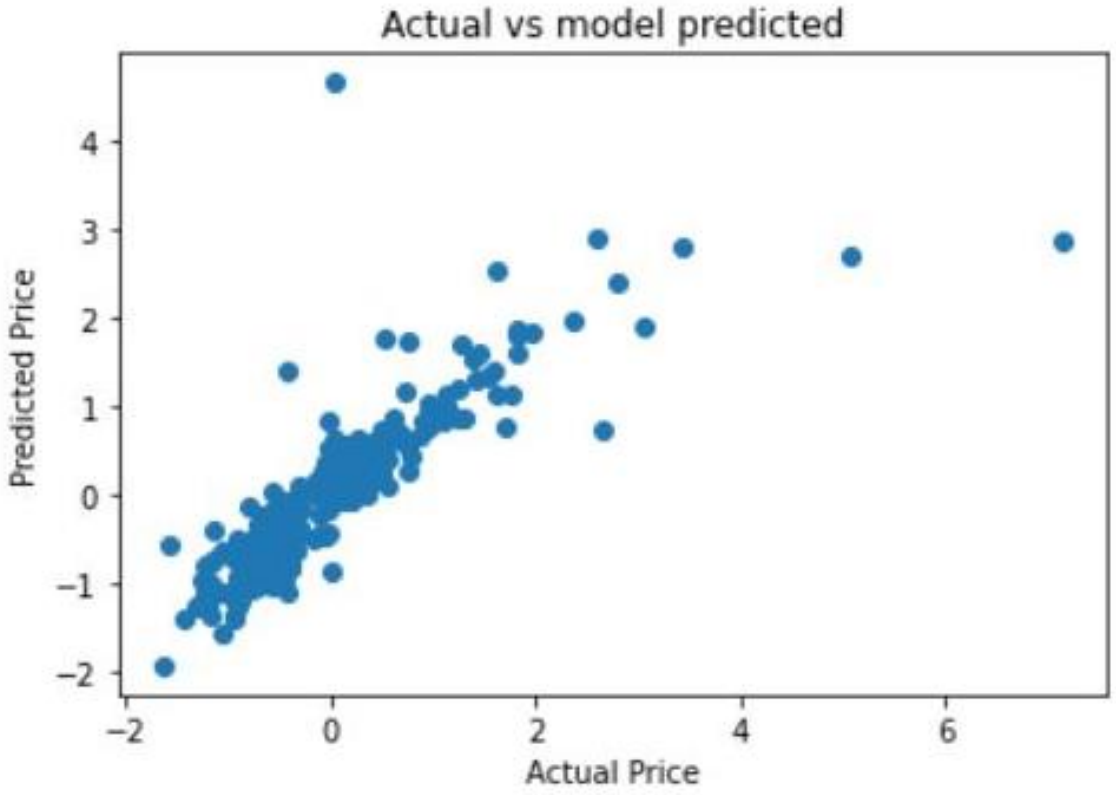
3. It will take a good way for the management to understand the pricing dynamics of a new project.

4. By doing visualization there are many things to be noted when it will according to work each other it means that from which aspect it is going to work either from sale type or condition as part from the sale price because it the main target of the housing case because according to this it will predicted whether it will rise in monthly or yearly.

5. By preprocessing the data it means that from the housing case label encoder helps the dataset column to transform to fit another column in to it.

6. Presumptions are by using regression label encoding, data scaling, that it means the relationship between the dependent and independent variables look fairly linear. Thus, our linearity assumption is satisfied.

MODEL DASHBOARD



FINALIZED MODEL:

the accuracy score was 86.36%

```
====================Train Result=====================
Accuracy Score: 99.01960784313727
_____________________________________________________
CLASSIFICATION REPORT :
              COD  CWD  ConLD  ConLw  New        WD  accuracy  \
precision 1.000000  1.0    1.0    1.0  1.0  0.989305  0.990196
recall    0.500000  1.0    1.0    1.0  1.0  1.000000  0.990196
f1-score  0.666667  1.0    1.0    1.0  1.0  0.994624  0.990196
support   4.000000  1.0    1.0    1.0 12.0 185.000000  0.990196

              macro avg  weighted avg
precision      0.998217      0.990301
recall         0.916667      0.990196
f1-score       0.943548      0.988588
support      204.000000    204.000000
_____________________________________________________
Cofusion Matrix:
[[  2   0   0   0   0   2]
 [  0   1   0   0   0   0]
 [  0   0   1   0   0   0]
 [  0   0   0   1   0   0]
 [  0   0   0   0  12   0]
 [  0   0   0   0   0 185]]


==================Test Result=======================
Accuracy: 86.36363636363636
_____________________________________________________
CLASSIFICATION REPORT :
              COD  New        WD  accuracy  macro avg  weighted avg
precision   0.0  0.0  0.938272  0.863636   0.312757      0.884961
recall      0.0  0.0  0.915663  0.863636   0.305221      0.863636
f1-score    0.0  0.0  0.926829  0.863636   0.308943      0.874169
support     1.0  4.0  83.000000  0.863636  88.000000     88.000000
_____________________________________________________
Confusion Matrix:
[[ 0  0  1]
 [ 0  0  4]
 [ 1  6 76]]
```

CONCLUSION

I would like to conclude here that this case is good way of predicting the data and its learning outcomes of the study in respect of Data Science from this it is very helpful in using visualization, data cleaning and various algorithms. By doing visualization there are many things to be noted when it will according to work each other it means that from which aspect it is going to work either from sale type or condition as part from the sale price because it the main target of the housing case because according to this it will predicted whether it will rise in monthly or yearly. By preprocessing the data it means that from the housing case label encoder helps the dataset column to transform to fit another column in to it. The model of the house price with the available independent variables and exactly the prices are vary with the variables.

THANK YOU