```python
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

house = pd.read_csv('train.csv')
```

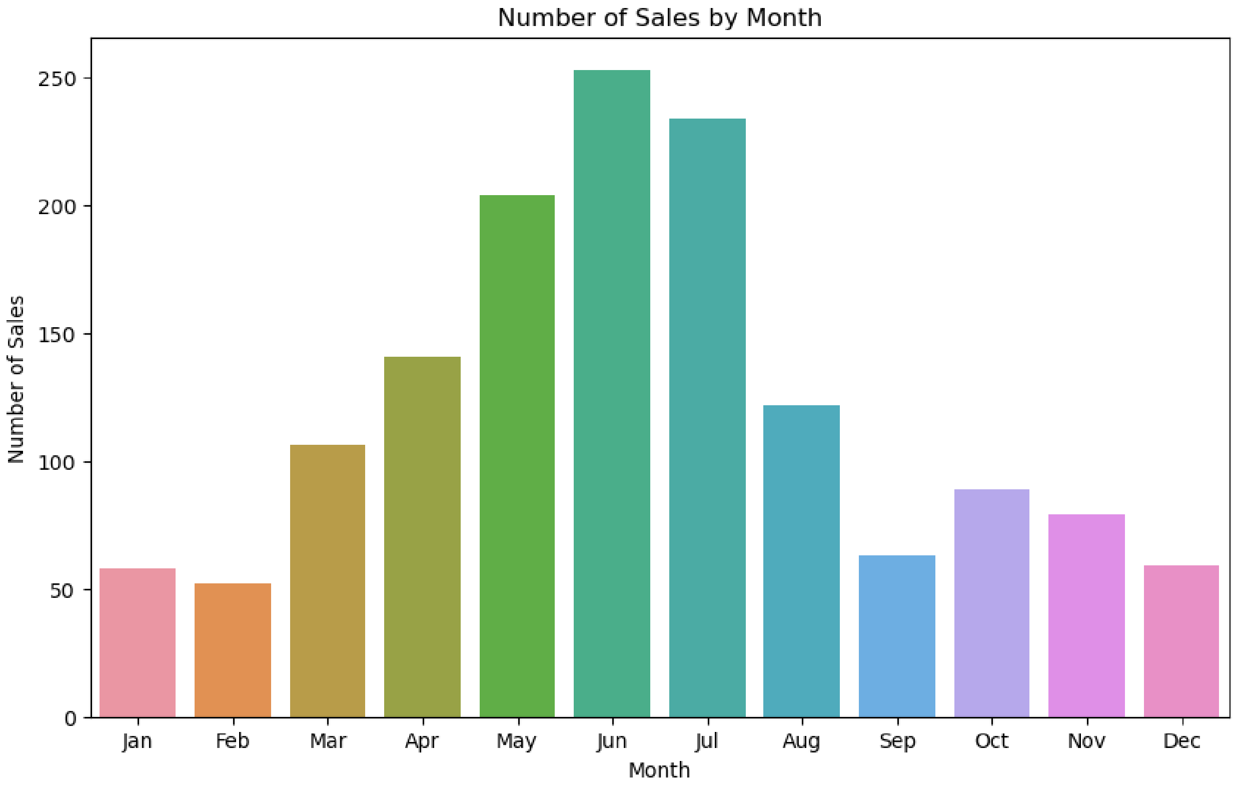
1- What is the month that the biggest amount of houses are sold? What is the month with the least amount?

```python
monthly_sales = house['MoSold'].value_counts().sort_index()
print(monthly_sales)
```

```
MoSold
1      58
2      52
3     106
4     141
5     204
6     253
7     234
8     122
9      63
10     89
11     79
12     59
Name: count, dtype: int64
```

```python
plt.figure(figsize=(10,6))
sns.barplot(x=monthly_sales.index, y=monthly_sales.values)
plt.xlabel('Month')
plt.ylabel('Number of Sales')
plt.title('Number of Sales by Month')
plt.xticks(ticks=range(0,12), labels=['Jan', 'Feb', 'Mar', 'Apr', 
'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
plt.show()
```

```
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498: 
FutureWarning: is_categorical_dtype is deprecated and will be removed 
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498: 
FutureWarning: is_categorical_dtype is deprecated and will be removed 
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498: 
FutureWarning: is_categorical_dtype is deprecated and will be removed 
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```



Answer: the months of June have the biggest amount of sales of houses. The month of February has the least. There is a significant increase of sales between March and August. Maybe due to the end of the Fiscal Year.

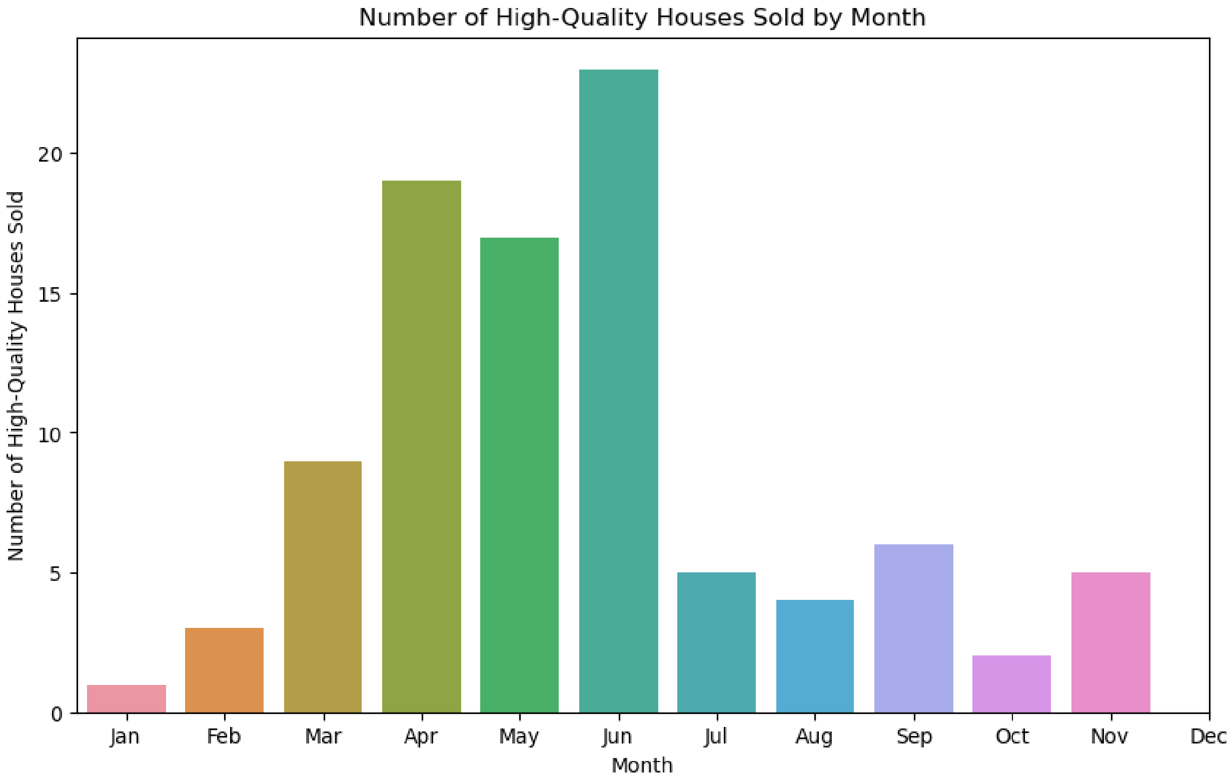
2. What is the month with the biggest amount of high quality houses sold (OverallCond: 8, 9, 10)?

```
high_quality_houses = house[house['OverallCond'].isin([8, 9, 10])]
hq_month_sales =
high_quality_houses['MoSold'].value_counts().sort_index()
print(hq_month_sales)

MoSold
2      1
3      3
4      9
5     19
6     17
7     23
8      5
9      4
10     6
11     2
12     5
Name: count, dtype: int64
```

```
plt.figure(figsize=(10,6))
sns.barplot(x=hq_month_sales.index, y=hq_month_sales.values)
plt.xlabel('Month')
plt.ylabel('Number of High-Quality Houses Sold')
plt.title('Number of High-Quality Houses Sold by Month')
plt.xticks(ticks=range(0,12), labels=['Jan', 'Feb', 'Mar', 'Apr',
'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
plt.show()
```

```
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```



Answer: the month of June have the biggest amount of sales of high quality houses. The high quality houses sales follows the house sales month distribution quite similarly.

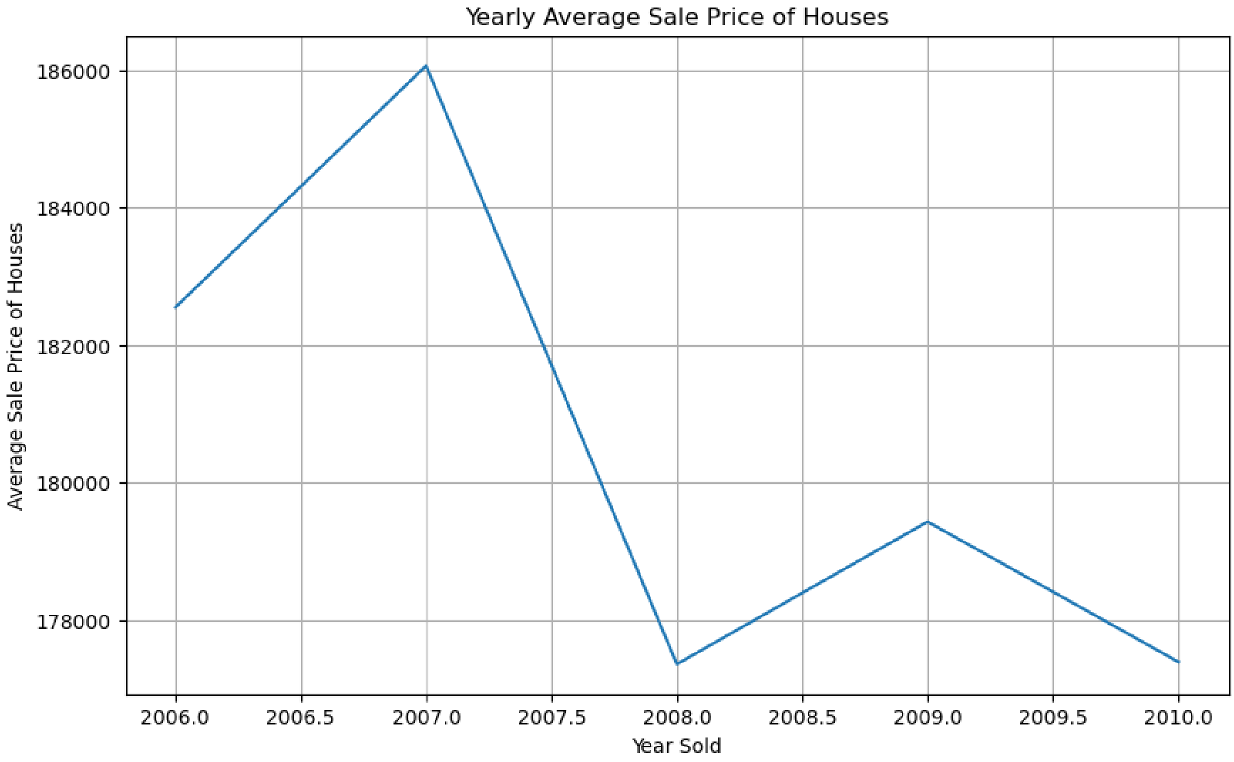
3. Does the year sold influence the price of houses?

```
avg_price_per_year = house.groupby('YrSold')['SalePrice'].mean()
print(avg_price_per_year)
```

```
YrSold
2006    182549.458599
2007    186063.151976
2008    177360.838816
2009    179432.103550
2010    177393.674286
Name: SalePrice, dtype: float64
```

```
plt.figure(figsize=(10,6))
sns.lineplot(x=avg_price_per_year.index, y=avg_price_per_year.values)
plt.xlabel('Year Sold')
plt.ylabel('Average Sale Price of Houses')
plt.title('Yearly Average Sale Price of Houses')
plt.grid(True)
plt.show()
```

```
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.
  with pd.option_context('mode.use_inf_as_na', True):
```

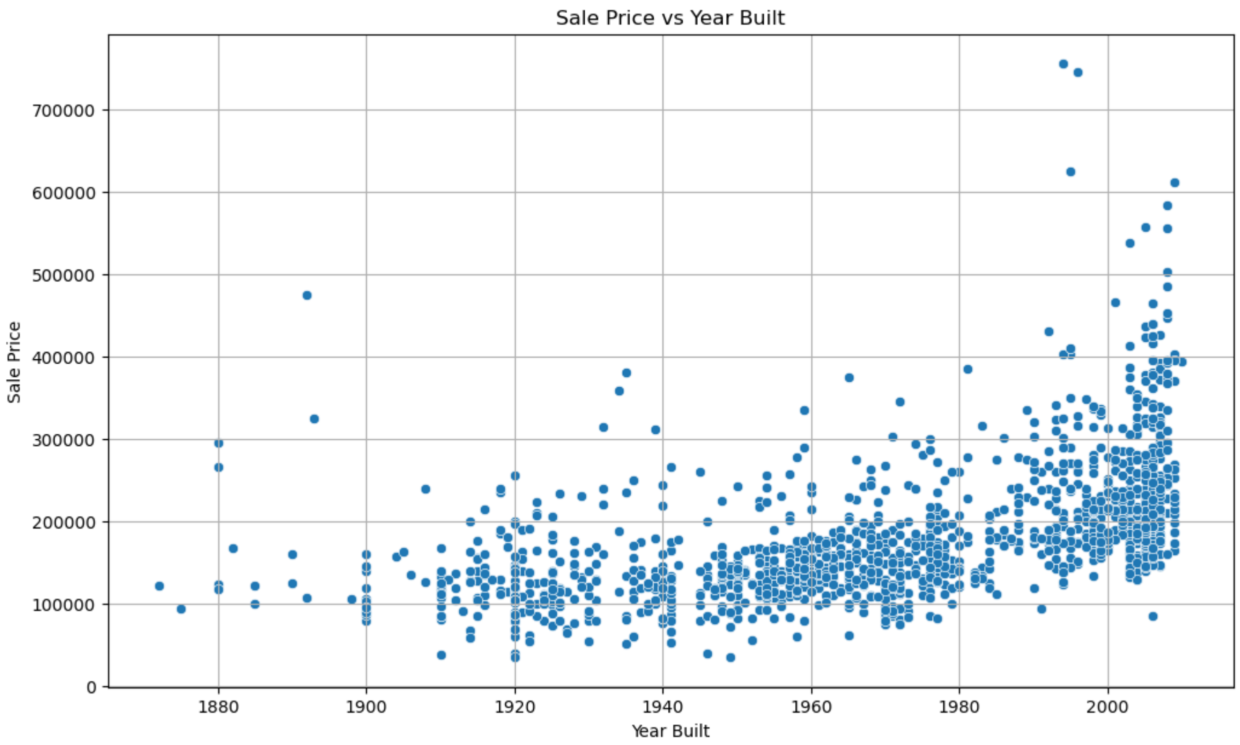


Answer: The price of houses went up until 2017 On 2008 it had a significant decrease

4. Does YearBuilt has any relation with SalePrice?

```
plt.figure(figsize=(12, 7))
sns.scatterplot(x=house['YearBuilt'], y=house['SalePrice'])
plt.title('Sale Price vs Year Built')
plt.xlabel('Year Built')
plt.ylabel('Sale Price')
plt.grid(True)
plt.show()
```

```
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```



```
correlation = house['YearBuilt'].corr(house['SalePrice'])
print(f"The correlation between YearBuilt and SalePrice is
{correlation:.2f}")

The correlation between YearBuilt and SalePrice is 0.52
```

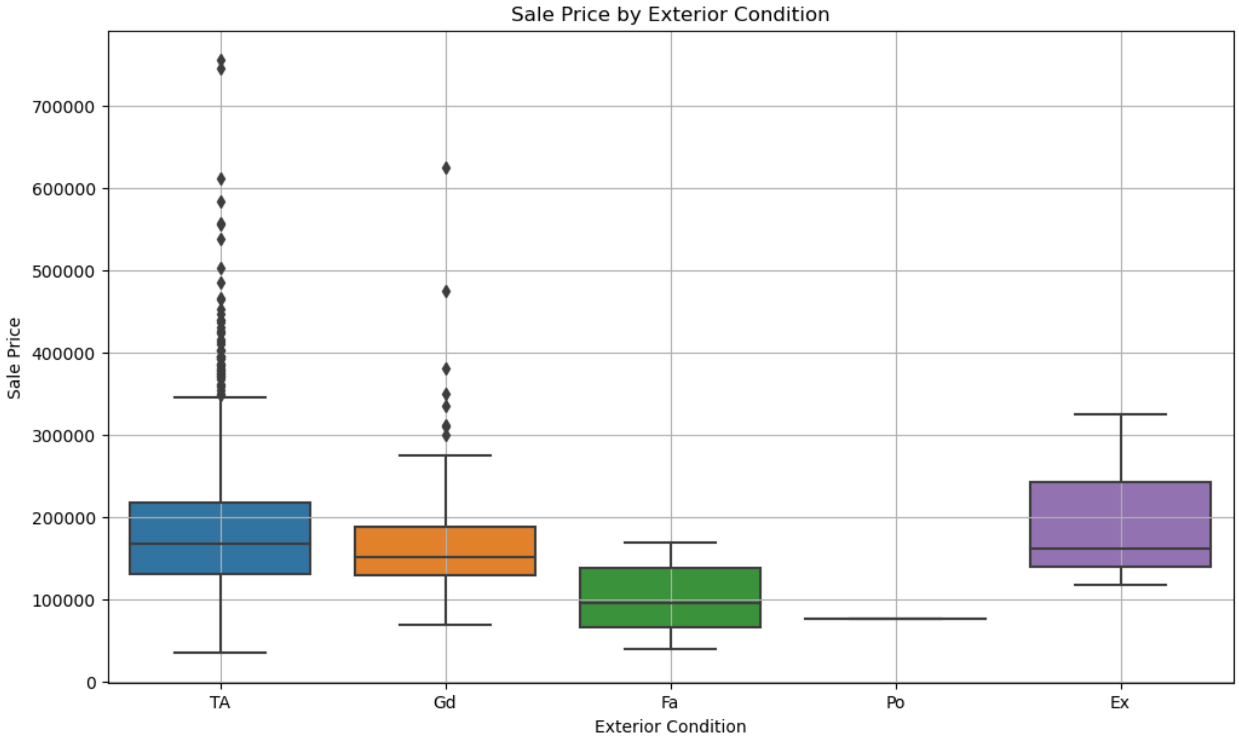
plt.figure(figsize=(12, 7)) sns.regplot(x=house['YearBuilt'], y=house['SalePrice'], scatter_kws={'s':10}, line_kws={'color':'red'}) plt.title('Regression Plot: Sale Price vs Year Built') plt.xlabel('Year Built') plt.ylabel('Sale Price') plt.grid(True) plt.show()

5. Does ExterCond has a relation with SalePrice?

```
plt.figure(figsize=(12, 7))
sns.boxplot(x=house['ExterCond'], y=house['SalePrice'])
plt.title('Sale Price by Exterior Condition')
plt.xlabel('Exterior Condition')
plt.ylabel('Sale Price')
plt.grid(True)
plt.show()

C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
```

```
  if pd.api.types.is_categorical_dtype(vector):
C:\Users\alefe\anaconda3\Lib\site-packages\seaborn\_oldcore.py:1498:
FutureWarning: is_categorical_dtype is deprecated and will be removed
in a future version. Use isinstance(dtype, CategoricalDtype) instead
  if pd.api.types.is_categorical_dtype(vector):
```



```python
grouped_data = house.groupby('ExterCond')['SalePrice'].describe()
print(grouped_data)
```

```
            count           mean            std       min       25%
50%  \
ExterCond

Ex            3.0  201333.333333  109235.220205  118000.0  139500.0
161000.0
Fa           28.0  102595.142857   40094.383940   39300.0   65500.0
95750.0
Gd          146.0  168897.568493   72608.303632   68400.0  128625.0
151250.0
Po            1.0   76500.000000            NaN   76500.0   76500.0
76500.0
TA         1282.0  184034.896256   79806.257233   34900.0  131100.0
167370.0

                 75%       max
ExterCond
```

```
Ex         243000.00  325000.0
Fa         137750.00  169500.0
Gd         187375.00  625000.0
Po          76500.00   76500.0
TA         217334.25  755000.0
```

The number of Excelent condition exteriors (Ex) is too small in relation to the other ones, the same applies tp the poor conditional exteriors (Po). The TA (Average/typical) conatains significantly more data, what makes hard to infer anny relationship between ExterCond and SalePrice

6. What's the neiborhood with the biggest Sale Price mean?

```
high_price_neiborhood = house.groupby('Neighborhood')
['SalePrice'].mean()
print(high_price_neiborhood)
```

```
Neighborhood
Blmngtn    194870.882353
Blueste    137500.000000
BrDale     104493.750000
BrkSide    124834.051724
ClearCr    212565.428571
CollgCr    197965.773333
Crawfor    210624.725490
Edwards    128219.700000
Gilbert    192854.506329
IDOTRR     100123.783784
MeadowV     98576.470588
Mitchel    156270.122449
NAmes      145847.080000
NPkVill    142694.444444
NWAmes     189050.068493
NoRidge    335295.317073
NridgHt    316270.623377
OldTown    128225.300885
SWISU      142591.360000
Sawyer     136793.135135
SawyerW    186555.796610
Somerst    225379.837209
StoneBr    310499.000000
Timber     242247.447368
Veenker    238772.727273
Name: SalePrice, dtype: float64
```

```
max_neiborhood = high_price_neiborhood.idxmax()
max_mean_price = high_price_neiborhood.max()
print(f'The neiborhood with the biggest mean sale price is
{max_neiborhood} with a mean price of ${max_mean_price:.2f}')
```

```
The neiborhood with the biggest mean sale price is NoRidge with a mean
price of $335295.32
```