In [11]:

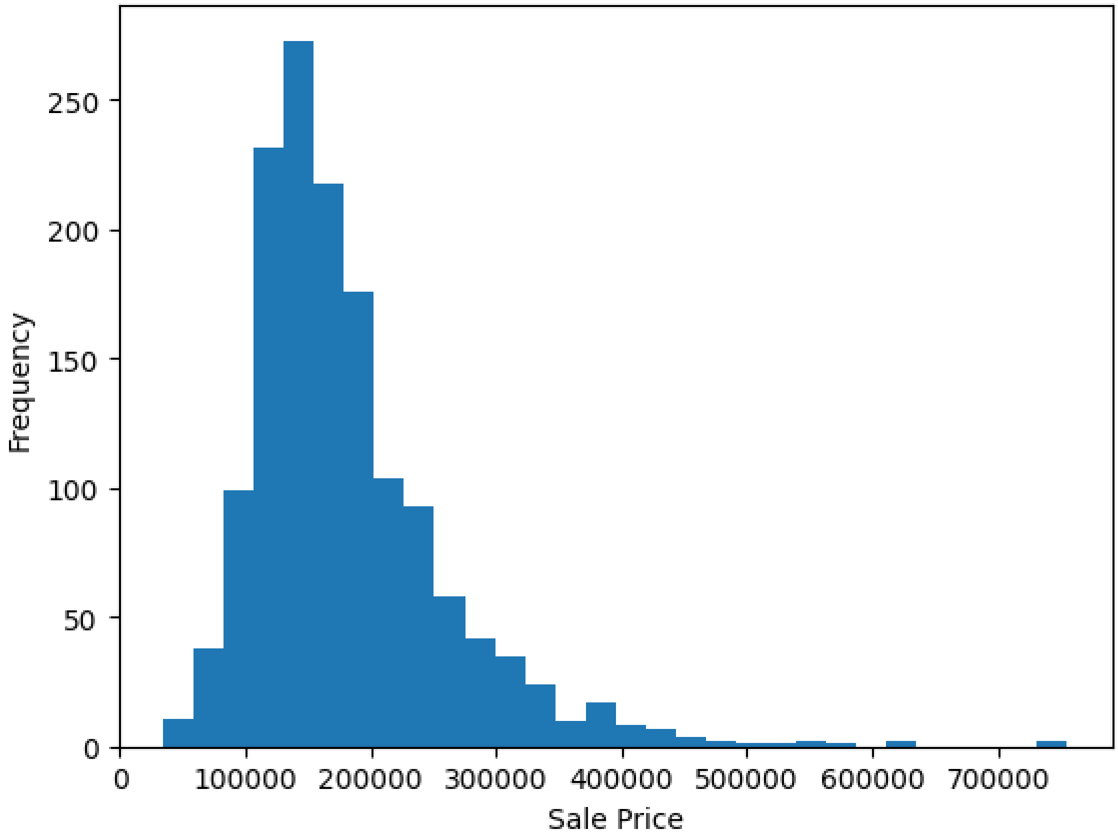
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [12]:

```
# Load the dataset
train = pd.read_csv('data/train.csv')

# Summary statistics
print(train.describe())
```

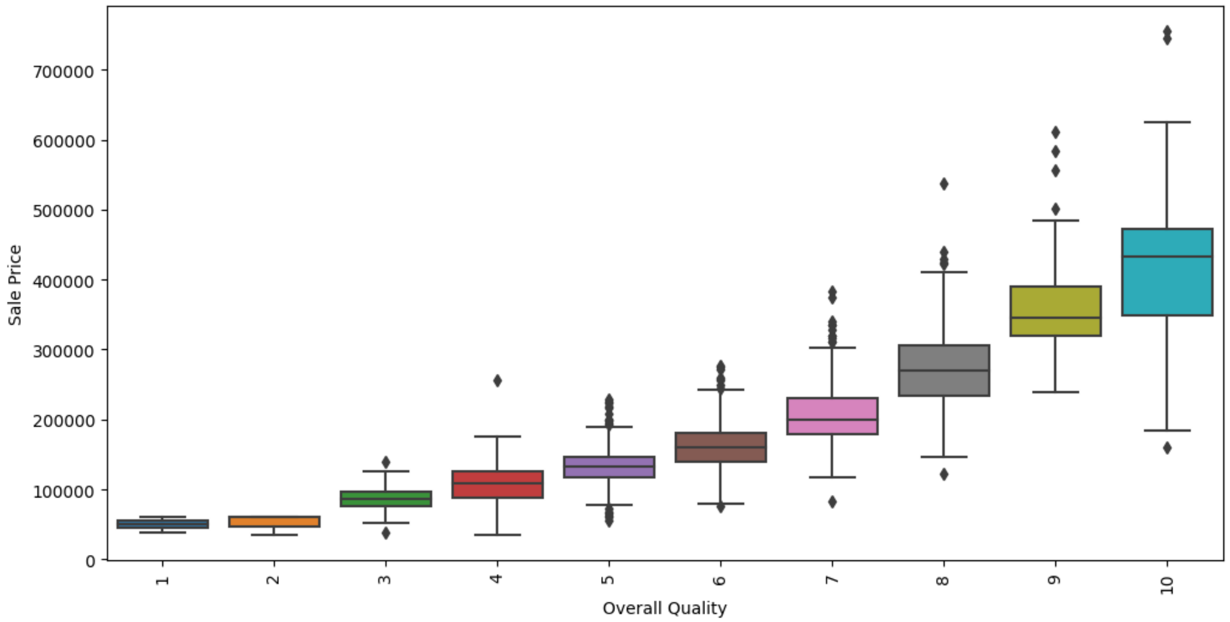
```
                Id   MSSubClass  LotFrontage        LotArea  OverallQual
\
count  1460.000000  1460.000000  1201.000000    1460.000000  1460.000000
mean    730.500000    56.897260    70.049958   10516.828082     6.099315
std     421.610009    42.300571    24.284752    9981.264932     1.382997
min       1.000000    20.000000    21.000000    1300.000000     1.000000
25%     365.750000    20.000000    59.000000    7553.500000     5.000000
50%     730.500000    50.000000    69.000000    9478.500000     6.000000
75%    1095.250000    70.000000    80.000000   11601.500000     7.000000
max    1460.000000   190.000000   313.000000  215245.000000    10.000000

       OverallCond    YearBuilt  YearRemodAdd   MasVnrArea   BsmtFinSF1
...  \
count  1460.000000  1460.000000   1460.000000  1452.000000  1460.000000
...
mean      5.575342  1971.267808   1984.865753   103.685262   443.639726
...
std       1.112799    30.202904     20.645407   181.066207   456.098091
...
min       1.000000  1872.000000   1950.000000     0.000000     0.000000
...
25%       5.000000  1954.000000   1967.000000     0.000000     0.000000
...
50%       5.000000  1973.000000   1994.000000     0.000000   383.500000
...
75%       6.000000  2000.000000   2004.000000   166.000000   712.250000
...
max       9.000000  2010.000000   2010.000000  1600.000000  5644.000000
...

        WoodDeckSF  OpenPorchSF  EnclosedPorch    3SsnPorch  ScreenPorch
\
count  1460.000000  1460.000000    1460.000000  1460.000000  1460.000000
mean     94.244521    46.660274      21.954110     3.409589    15.060959
std     125.338794    66.256028      61.119149    29.317331    55.757415
min       0.000000     0.000000       0.000000     0.000000     0.000000
25%       0.000000     0.000000       0.000000     0.000000     0.000000
50%       0.000000    25.000000       0.000000     0.000000     0.000000
75%     168.000000    68.000000       0.000000     0.000000     0.000000
max     857.000000   547.000000     552.000000   508.000000   480.000000

          PoolArea       MiscVal       MoSold       YrSold      SalePrice
count  1460.000000   1460.000000  1460.000000  1460.000000    1460.000000
mean      2.758904     43.489041     6.321918  2007.815753  180921.195890
std      40.177307    496.123024     2.703626     1.328095   79442.502883
min       0.000000      0.000000     1.000000  2006.000000   34900.000000
25%       0.000000      0.000000     5.000000  2007.000000  129975.000000
50%       0.000000      0.000000     6.000000  2008.000000  163000.000000
75%       0.000000      0.000000     8.000000  2009.000000  214000.000000
max     738.000000  15500.000000    12.000000  2010.000000  755000.000000

[8 rows x 38 columns]
```

```
In [13]: # Histogram of the target variable
         plt.hist(train['SalePrice'], bins=30)
         plt.xlabel('Sale Price')
         plt.ylabel('Frequency')
         plt.show()
```



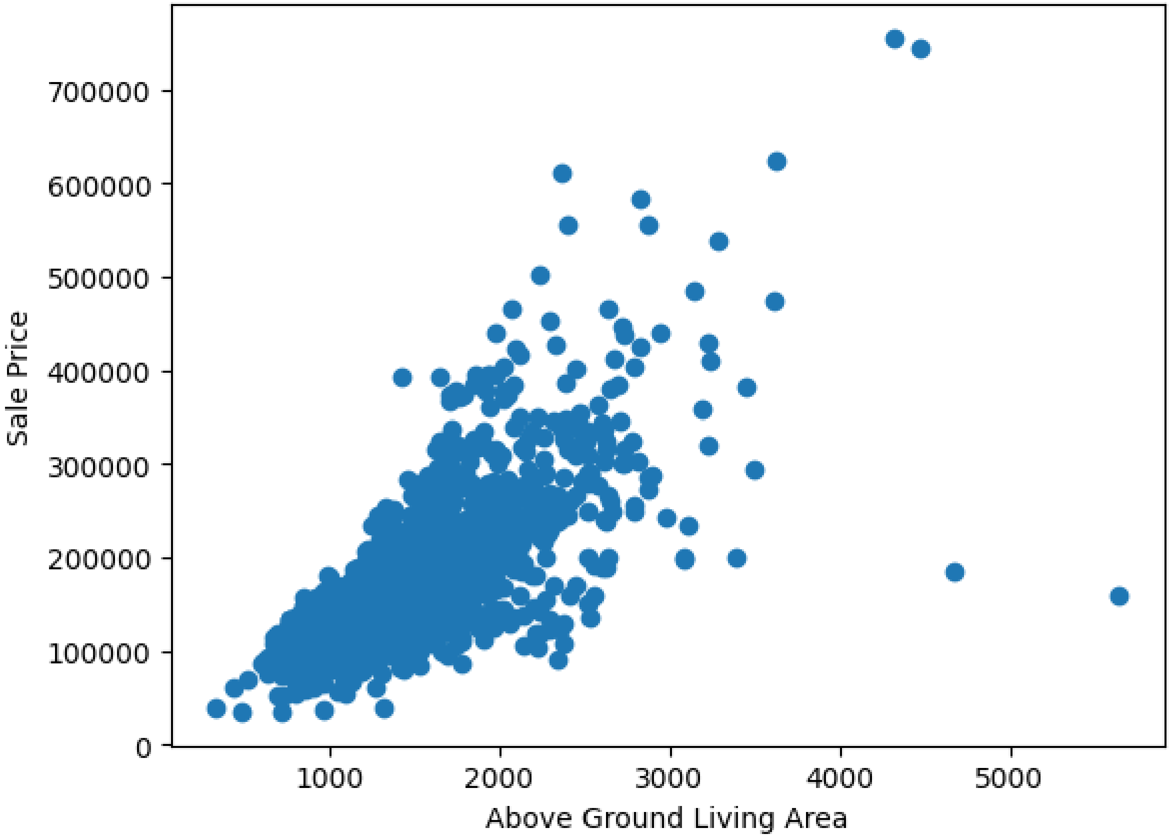
In [9]:

```
# Box plot of the target variable vs. a categorical feature
plt.figure(figsize=(12,6))
plt.xticks(rotation=90)
sns.boxplot(x='OverallQual', y='SalePrice', data=train)
plt.xlabel('Overall Quality')
plt.ylabel('Sale Price')
plt.show()
```

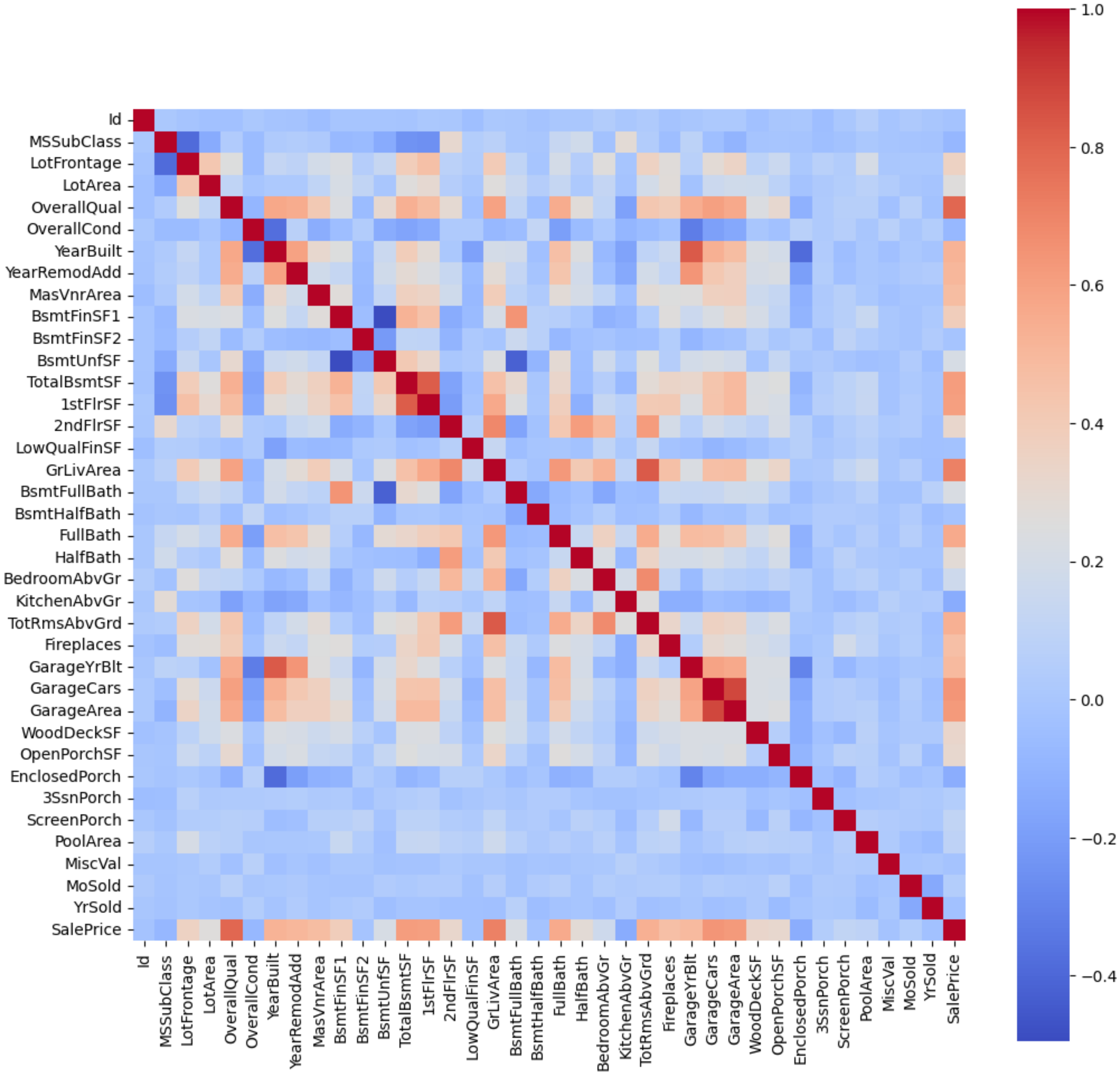


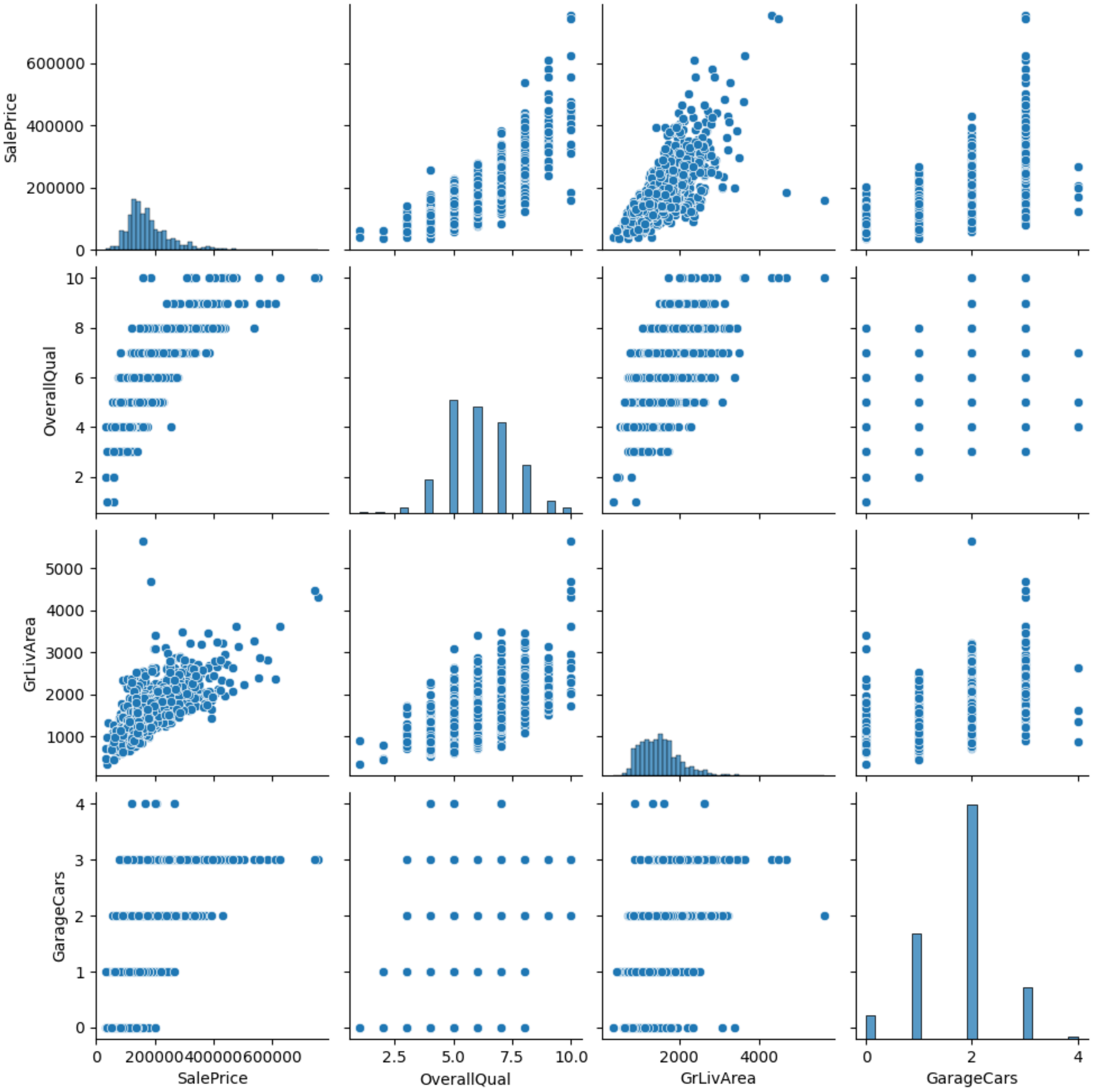
In [10]:

```
# Scatter plot of a continuous feature vs. the target variable
plt.scatter(train['GrLivArea'], train['SalePrice'])
plt.xlabel('Above Ground Living Area')
plt.ylabel('Sale Price')
plt.show()
```



```
/var/folders/zl/dmfc7n5j1656b9c_m5q7q2z80000gn/T/ipykernel_20541/32754248
96.py:8: FutureWarning: The default value of numeric_only in DataFrame.co
rr is deprecated. In a future version, it will default to False. Select o
nly valid columns or specify the value of numeric_only to silence this wa
rning.
  corr_matrix = train.corr()
```


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
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4



In [ ]:

```
# Heatmap of correlations between features
corr_matrix = train.corr()
plt.figure(figsize=(12,12))
sns.heatmap(corr_matrix, square=True, cmap='coolwarm')
plt.show()
```

In [ ]:

```
# Pairplot of select features
sns.pairplot(train[['SalePrice', 'OverallQual', 'GrLivArea', 'GarageCars']]
plt.show()
```