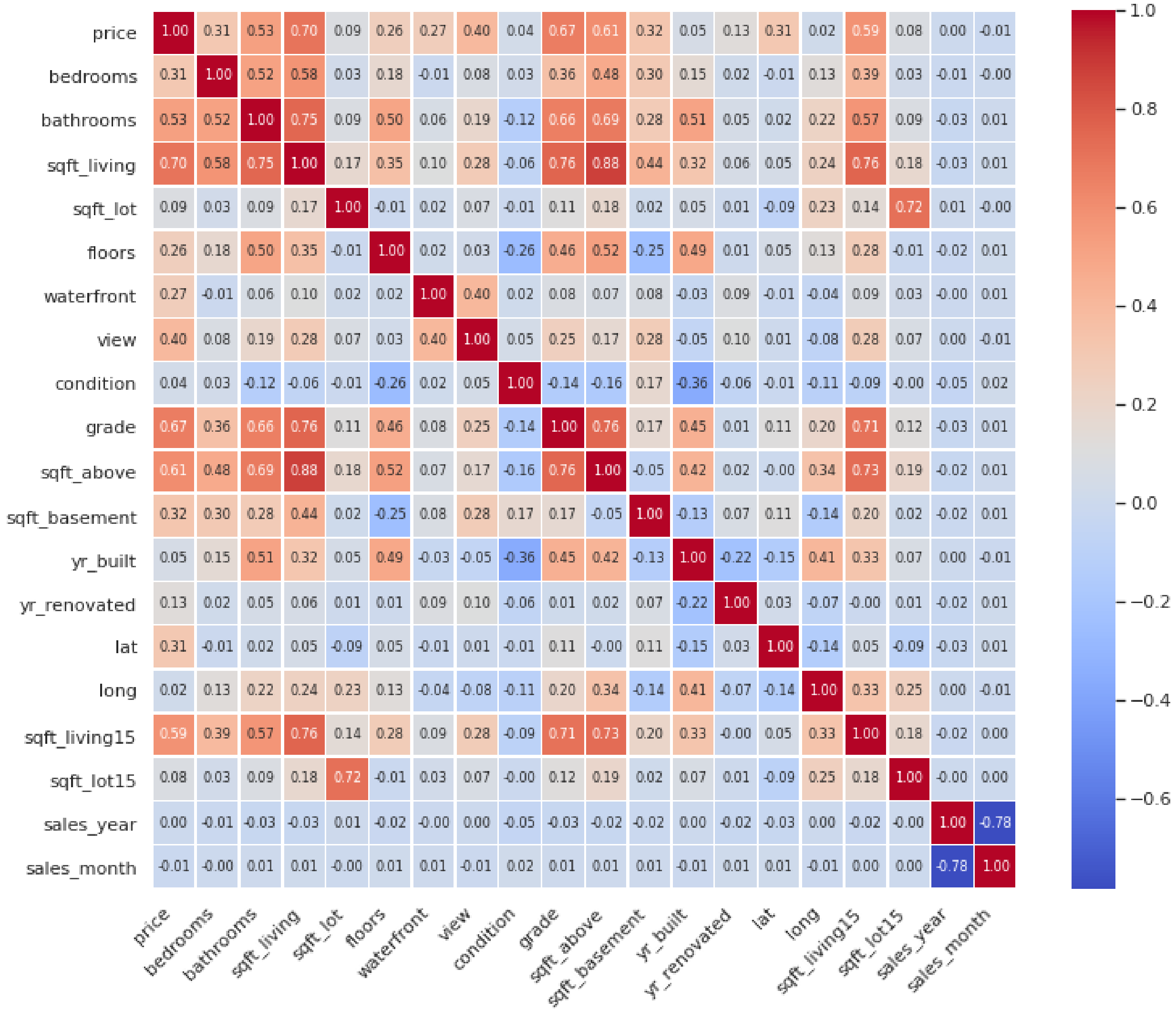
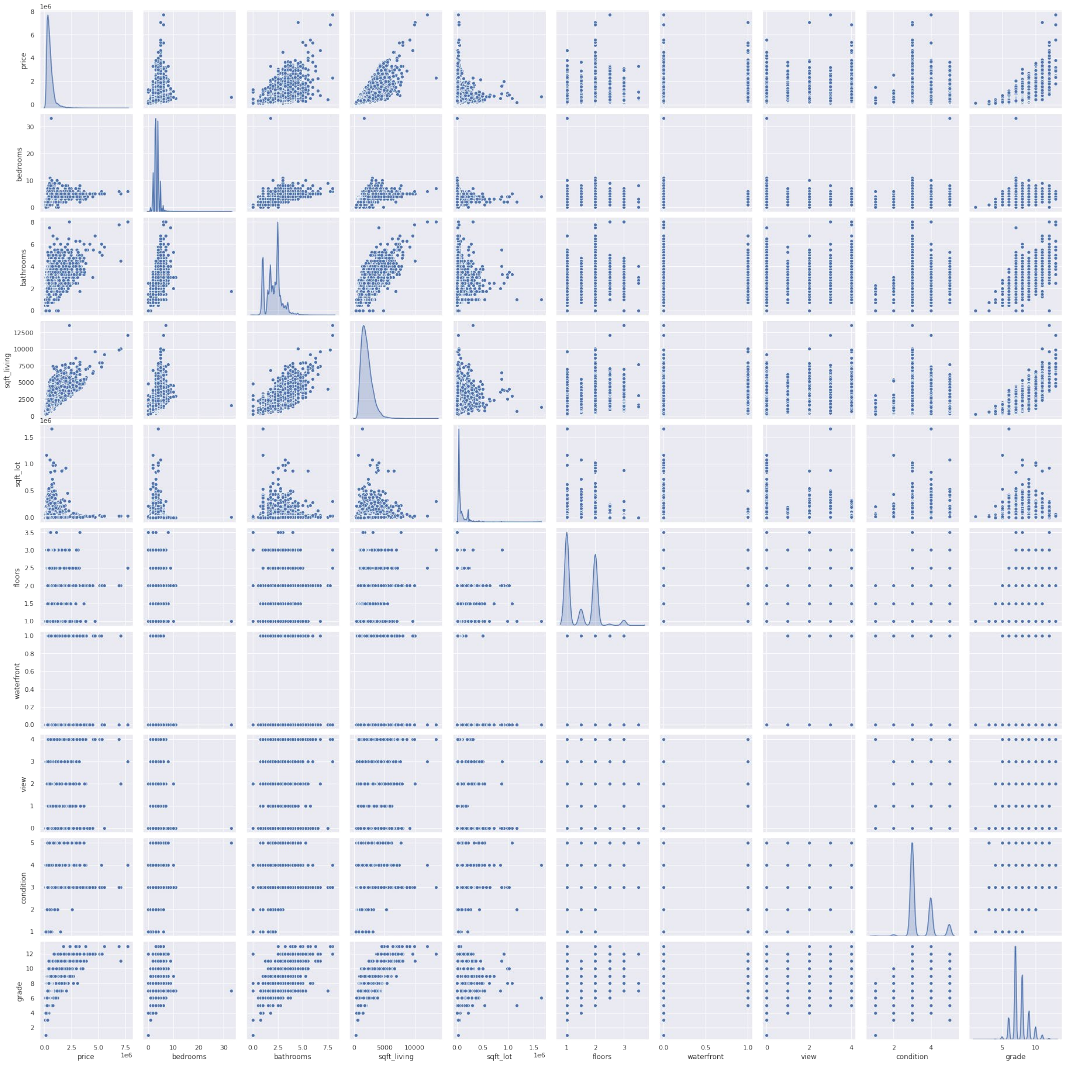
Data Cleaning and EDA



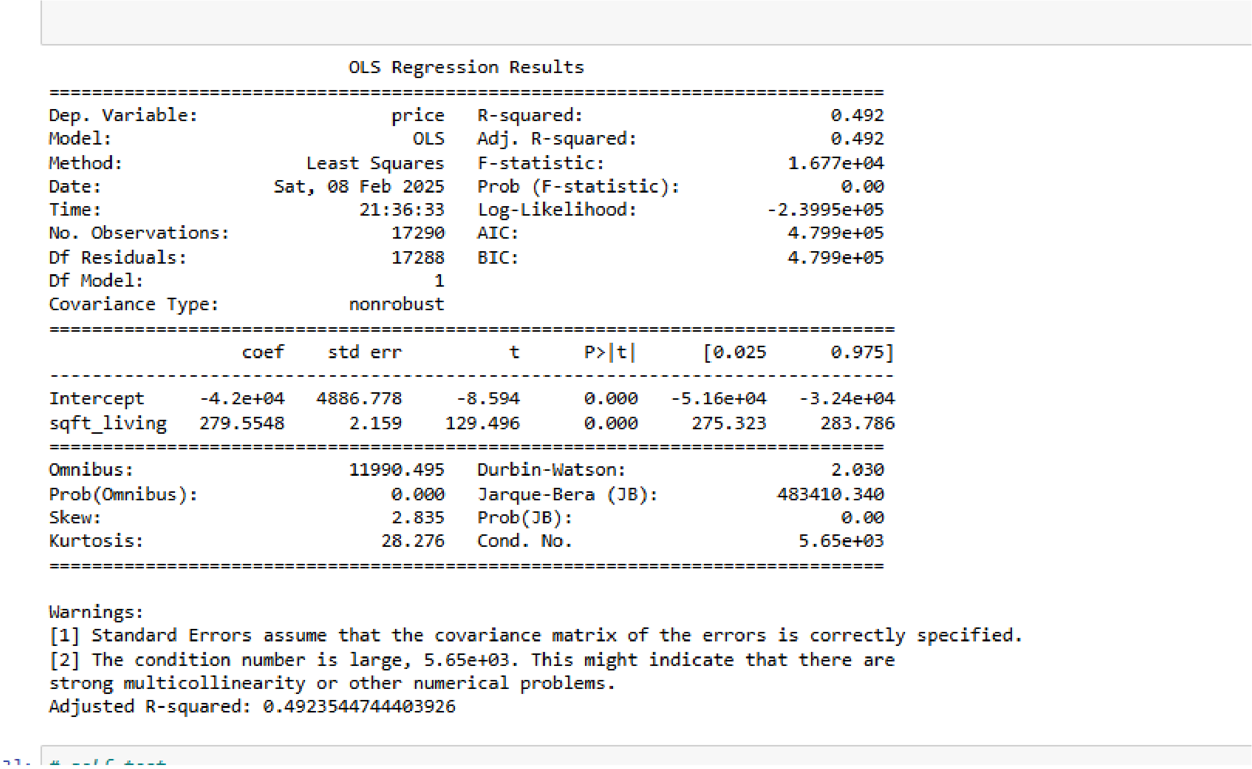


```
from sklearn.model_selection import train_test_split

# Split the data into training (80%) and testing (20%) while keeping 'price' in both
X_train, X_test = train_test_split(df, test_size=0.2, random_state=42)  # Set random_state for reproducibility

# Print the number of rows in each dataset
print("Training set size:", len(X_train))
print("Test set size:", len(X_test))
```

```
Training set size: 17290
Test set size: 4323
```



```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                  price   R-squared:                       0.492
Model:                            OLS   Adj. R-squared:                  0.492
Method:                 Least Squares   F-statistic:                 1.677e+04
Date:                Sat, 08 Feb 2025   Prob (F-statistic):               0.00
Time:                        21:36:33   Log-Likelihood:            -2.3995e+05
No. Observations:               17290   AIC:                         4.799e+05
Df Residuals:                   17288   BIC:                         4.799e+05
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
===============================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------
Intercept     -4.2e+04   4886.778     -8.594      0.000   -5.16e+04   -3.24e+04
sqft_living   279.5548      2.159    129.496      0.000     275.323     283.786
==============================================================================
Omnibus:                    11990.495   Durbin-Watson:                   2.030
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           483410.340
Skew:                           2.835   Prob(JB):                         0.00
Kurtosis:                      28.276   Cond. No.                     5.65e+03
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 5.65e+03. This might indicate that there are
strong multicollinearity or other numerical problems.
Adjusted R-squared: 0.4923544744403926
```

List:

sqft_living:	0.4923
grade:	0.4451
sqft_above:	0.3856
bathrooms:	0.3338
sqft_living15:	0.3249
view:	0.1877
sqft_basement:	0.1721
lat:	0.1124
waterfront:	0.1056
floors:	0.0895
yr_built:	0.0521
sqft_lot:	0.0087
sqft_lot15:	0.0079
yr_renovated:	0.0032
long:	0.0028
condition:	0.0019

The top 3 predictors based on the adj R squared: sqft_living , grade, and sqft_above. I conducted simple linear regression for each predictor using statsmodels.ols(), with price as the dependent variable. After fitting the models, I extracted the Adjusted R-squared values for each predictor to evaluate their performance. Finally, I ranked the predictors in descending order based on their Adjusted R-squared values to determine which variables had the strongest relationship with price. Yes, the correlation matrix analysis identified sqft_living as the best guess predictor because it had the highest correlation with price. After validating this through Adjusted R-squared values from linear regression, sqft_living remains the strongest predictor, confirming its significant relationship with price.