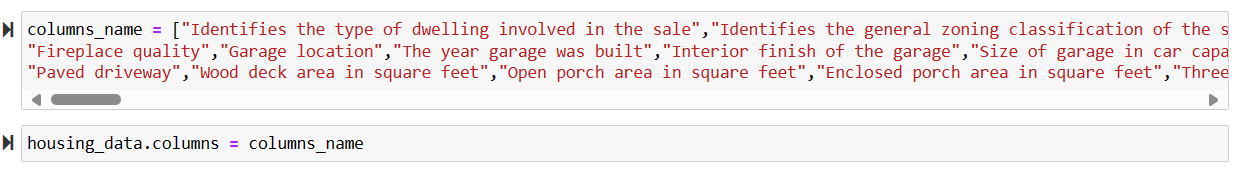
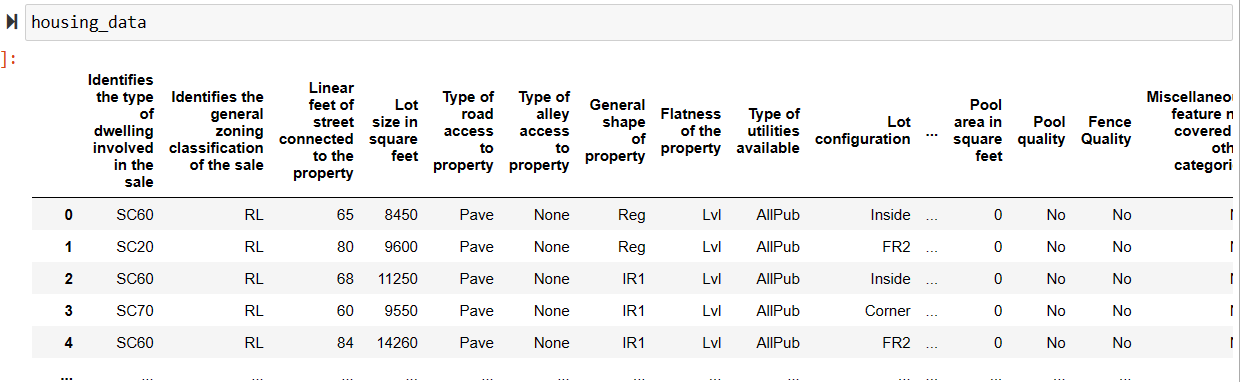
**Interim Submission Report**:

1. **Introduction:** I am pleased to present the interim submission of our House Price Prediction Module, a crucial component of our ongoing project aimed at developing an advanced real estate analytics system. This module plays a pivotal role in forecasting and estimating house prices, providing valuable insights for both homebuyers and sellers in the dynamic real estate market.

The first thing to do make data look very clear and good to analyse and visualize (here we changed column names and element names as per given dictionary to understand the project in very simple way) so after that with the visualization I can understand data and take meaningful from insights.

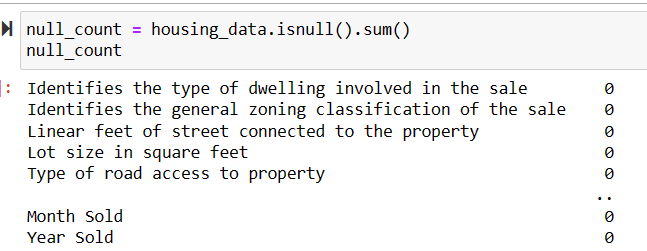




1. **Project Overview:** Providing a meaningful overview of the Real estate house sale price prediction project.

The House Sale Price Prediction project aims to develop a sophisticated predictive model for estimating residential property prices. Leveraging machine learning algorithms and comprehensive datasets, the project seeks to provide accurate and reliable predictions for potential homebuyers and sellers. Key components include data collection and cleaning, feature engineering, and the implementation of advanced machine learning techniques. The interim submission marks successful progress, with achieved milestones in data processing, model development, and preliminary testing. The next phases involve continuous refinement, integration with the broader real estate analytics system, and the development of a user-friendly interface. The ultimate goal is to deliver a robust and effective House Sale Price Prediction Module that enhances decision-making in the real estate market.

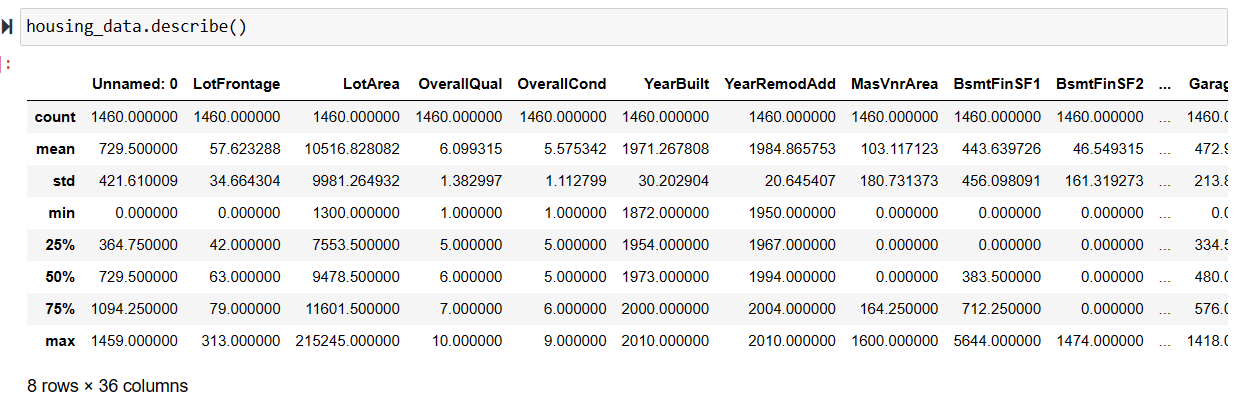
**Finding missing values:**

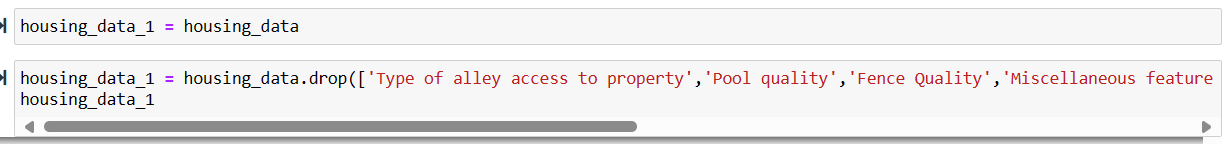
****

**Finding Shape of the data:**

****

**Describe of data:**

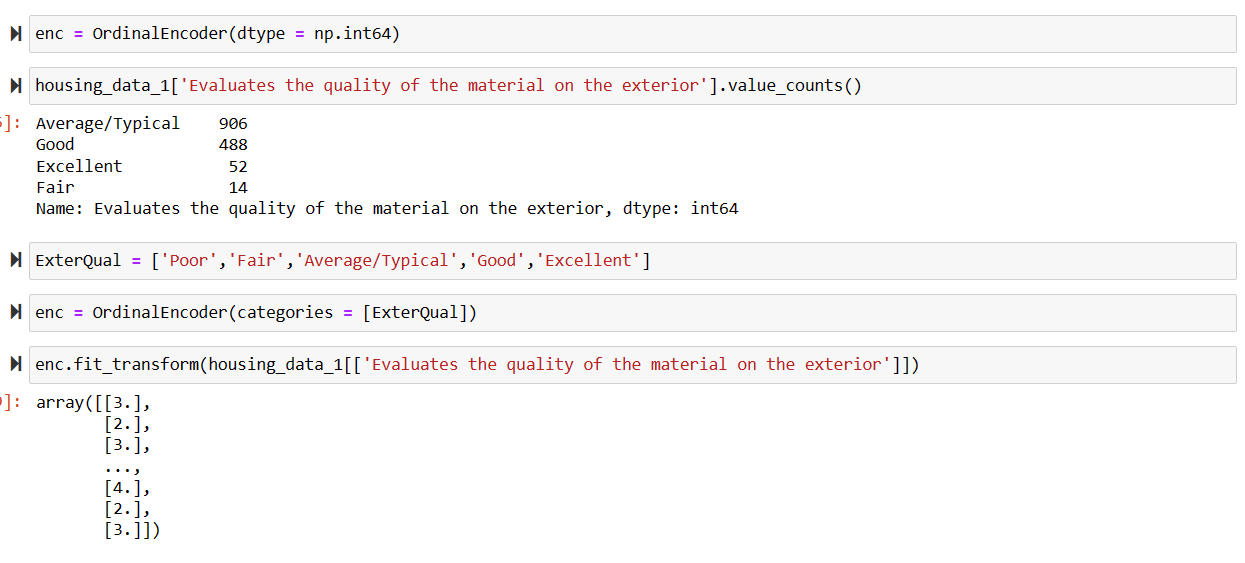
**Dropping Unwanted Columns:**

1. **Finding Duplicates:3. Key Findings:** Successful collection and cleaning of diverse datasets, ensuring the inclusion of various factors influencing house prices, such as historical transactions, economic indicators, and geographical features.

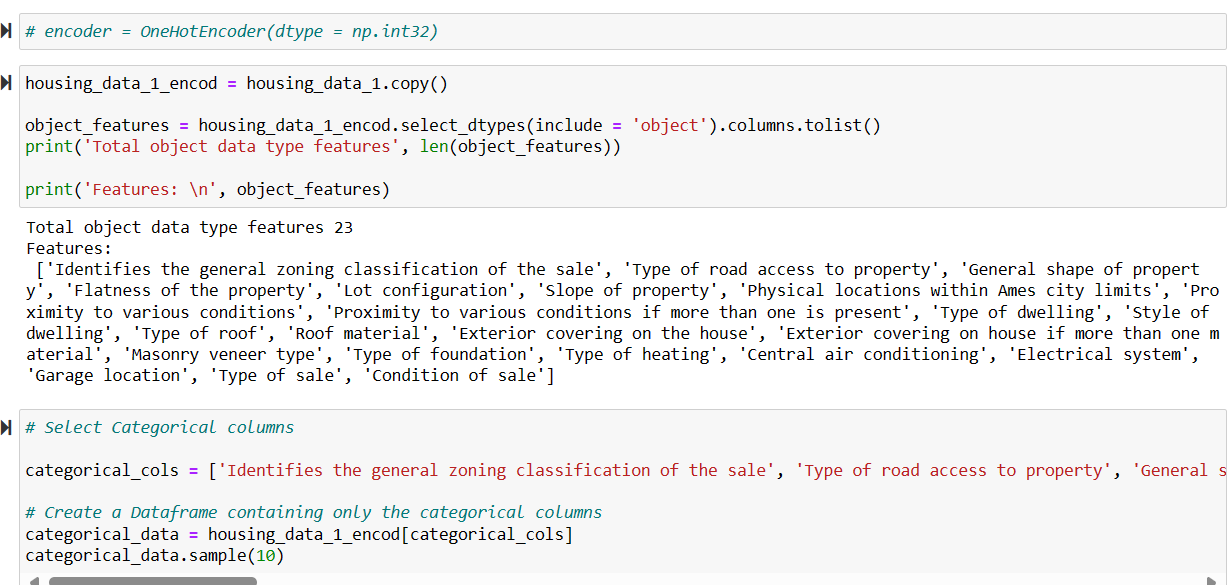
After Done such feature engineering part we have right now cleaned and good data to visualise and analyse some useful insights of the project like Locations, amenities, Property type, square foot, quality of the material, and with this type of features it’s time to understand what was the impact pattern to affect the Sale price.

1. **Significant Features:** Identification of key features strongly correlated with house prices, such as location, square footage, number of bedrooms/bathrooms, and amenities, indicating their importance in influencing property values.Top of Form
2. **Outliers and Anomalies:** Identification of outliers or anomalies in the dataset that might skew predictions, prompting the need for outlier handling techniques to improve model robustness.
3. **Feature Importance:** Ranking of features based on their impact on house prices, aiding in the prioritization of influential factors during model development and refinement.
4. **Feature Importance Ranked:** Features have been ranked based on their importance in influencing house prices, providing valuable insights for refining the model and focusing on the most impactful variables.
5. **Insights Extracted:** Actionable insights have been extracted from the data, offering valuable information for decision-makers in the real estate domain regarding market trends, investment opportunities, and potential areas for improvement.
6. **Facing challenges:** Predicting house prices is a complex task that involves various challenges. Here's a small overview of some common challenges faced during a house price prediction project.
7. **Data Quality and Quantity:**Limited and incomplete data can hinder the accuracy of predictions.Missing or erroneous data, such as inaccurate property details or outdated information, can affect model performance.
8. **Feature Selection and Engineering:**Identifying and selecting relevant features can be challenging, as some variables may have a nonlinear or complex relationship with house prices.Feature engineering is crucial for creating meaningful predictors from available data, but it requires domain knowledge and creativity.
9. **Outliers and Anomalies:**Outliers, such as extremely high or low-priced properties, can distort the model's training process.Robust methods are needed to handle outliers appropriately and prevent them from unduly influencing the predictions.
10. **Temporal Dynamics:**Housing markets are dynamic, and factors influencing prices can change over time.The model needs to account for seasonality, economic trends, and other temporal variations to make accurate predictions.
11. **Finding Outliers:** Finding outliers is very impressive task but before trimming and caping outliers we want to change categorical data into numerical data with the help of **Ordinal Encoding and One Hot Encoding** for better dataset understanding.

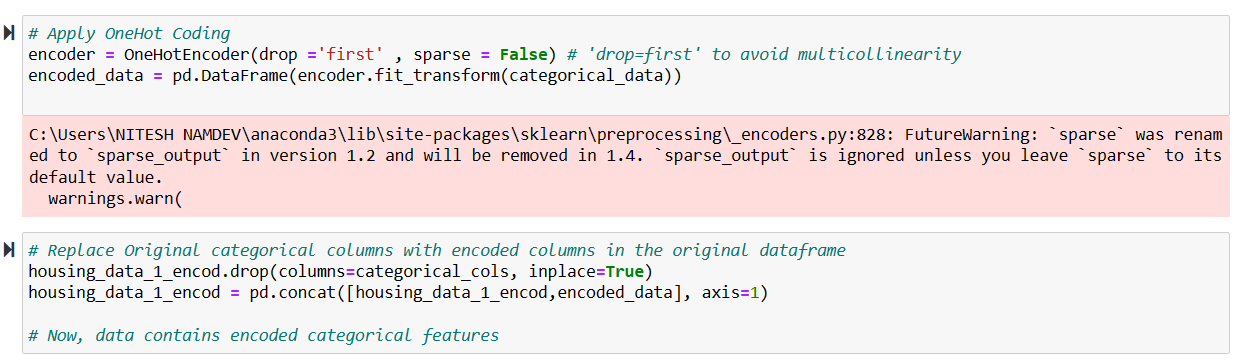
Ordinal Encoding: 



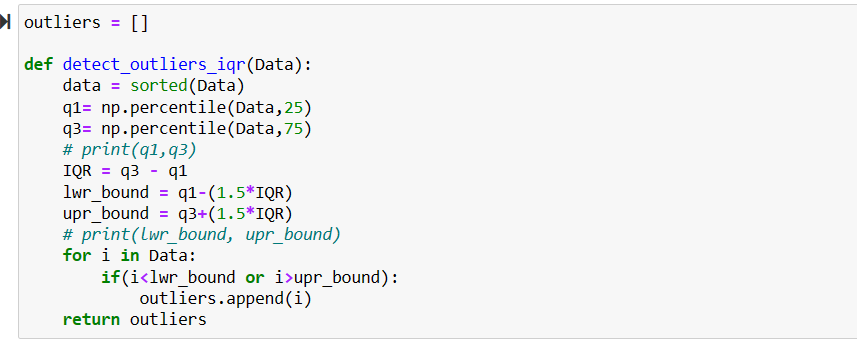
One Hot Encoding:

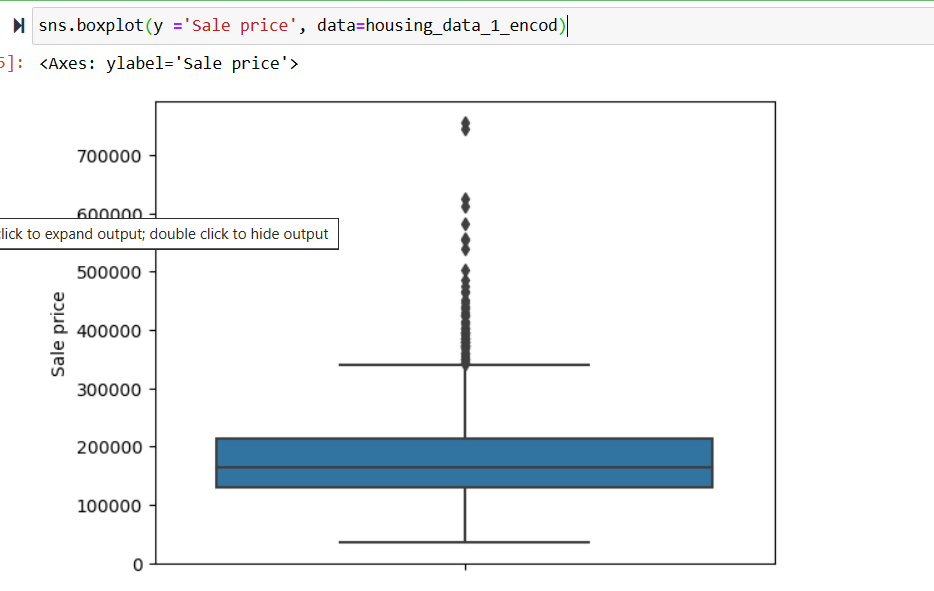


1. The code snippet involves the following steps: Encoding Using One Hot Encoder: The One Hot Encoder is used to encode categorical features in the housing data. Copying the Dataset: A copy of the original housing dataset is created. Selecting Object Data Type Features: The code identifies features that are of object data type (categorical features). Creating a Data Frame with Categorical Columns: A new Data Frame is created containing only the categorical columns.2. The total number of object data type features is 23.3. The code snippet demonstrates how to handle categorical data efficiently in Python using Pandas.



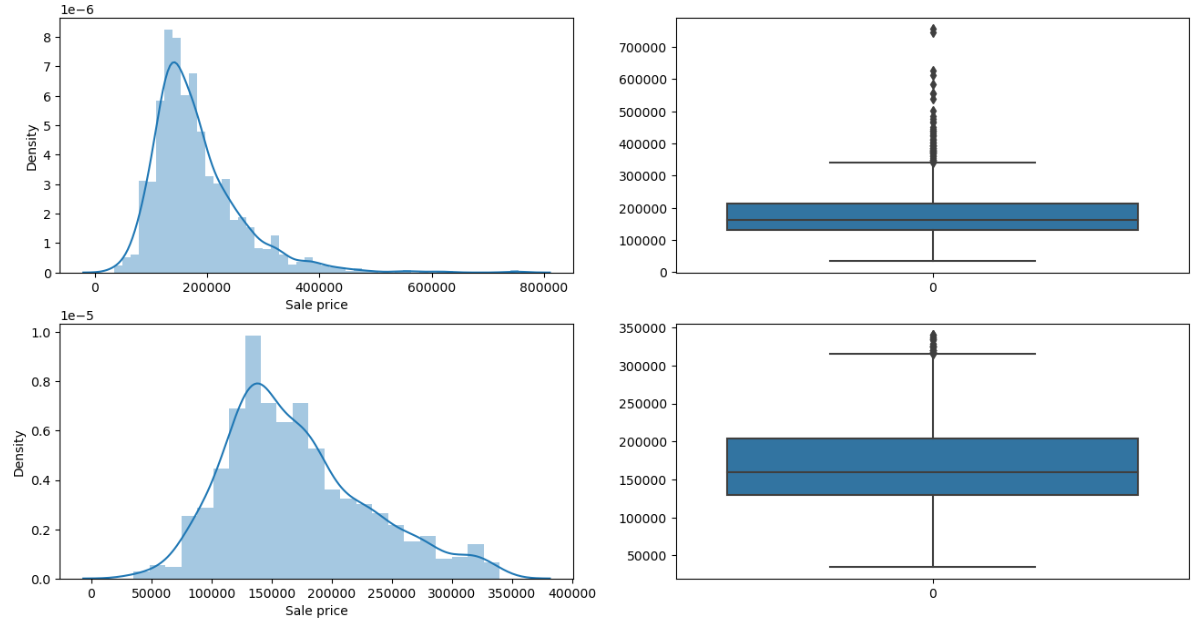
The code snippet involves the following steps:Encoding Using One Hot Encoder: The One Hot Encoder is used to encode categorical features in the housing data.Copying the Dataset: A copy of the original housing dataset is created.Selecting Object Data Type Features: The code identifies features that are of object data type (categorical features).Creating a Data Frame with Categorical Columns: A new Data Frame is created containing only the categorical columns.The total number of object data type features is 23.The code snippet demonstrates how to handle categorical data efficiently in Python using Pandas.

Finding Outliers: 



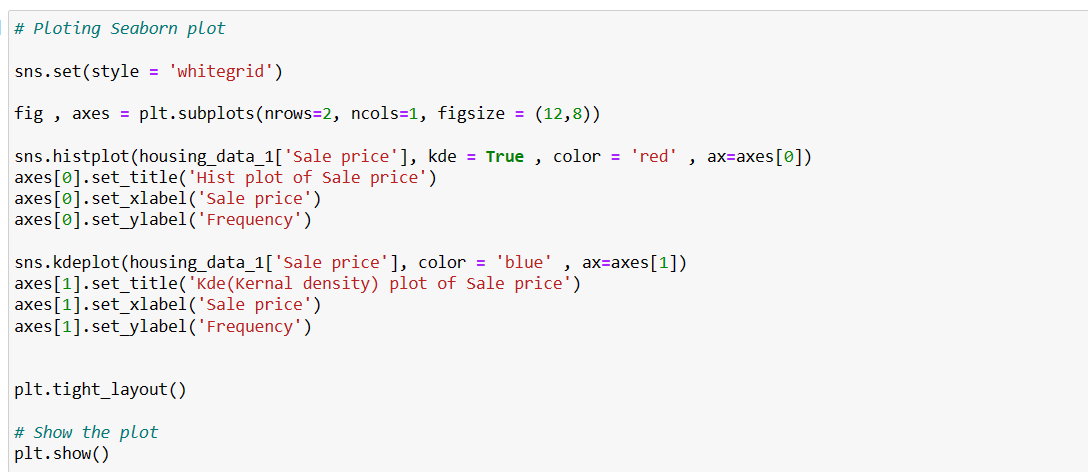
Boxplot Description:The graph displays a range of housing sale prices.The y-axis represents the Sale Price in an unspecified currency.The central blue box represents the interquartile range (IQR), which contains the middle 50% of sale prices.Outliers (represented by black dots) lie outside the IQR and indicate significantly higher-priced houses.Outliers:The presence of outliers suggests that there are houses with exceptionally high sale prices.These outliers could be luxury properties, unique estates, or other exceptional cases.Python Code:The graph was generated using Python, specifically the seaborn library.The code snippet at the top indicates the use of sns.boxplot(y='Sale price', data=housing\_data\_1\_encod).

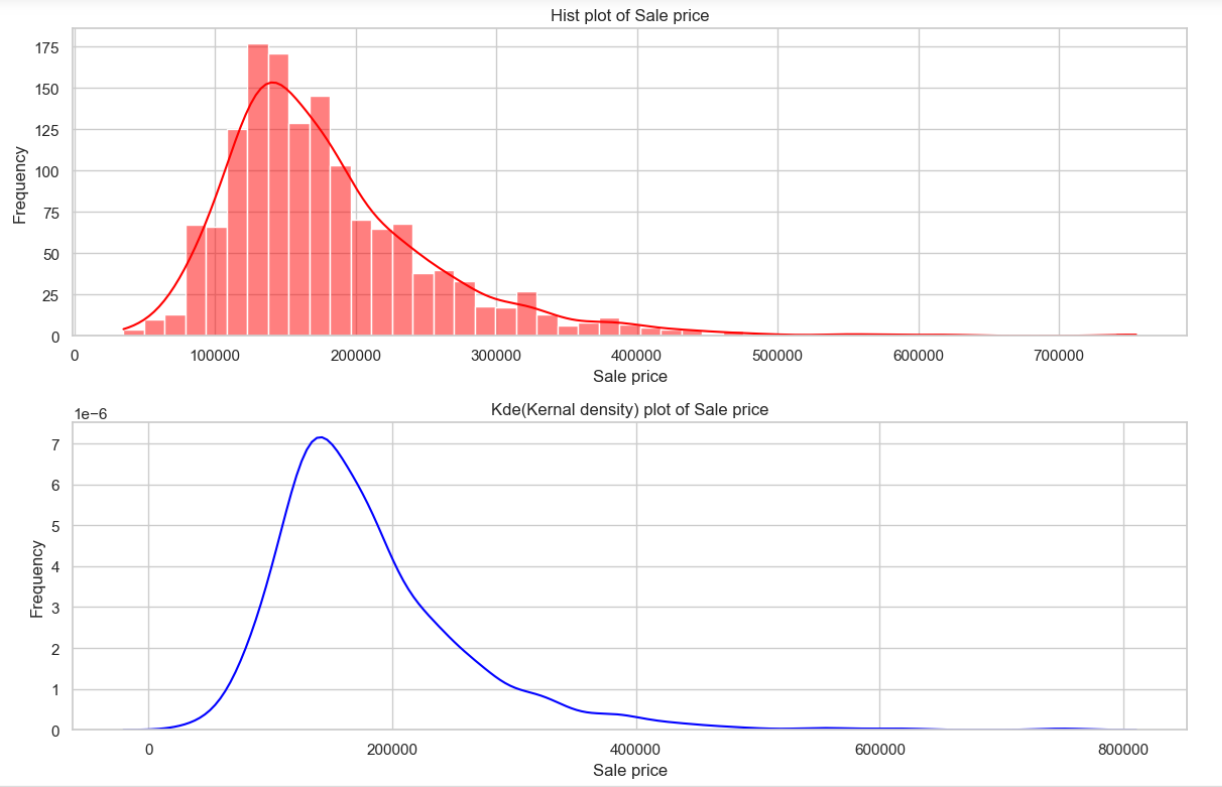
**With Or Without Outliers:**



Density Plots:The top-left plot shows a density distribution of sale prices ranging from 0 to approximately 800,000. The highest density occurs around 200,000.The bottom-left plot represents another dataset with sale prices ranging from 0 to about 350,000. The peak density is near 150,000.Both density plots exhibit positive skewness, indicating that most data points are concentrated on the lower end of the sale price spectrum.Box Plots:The top-right plot corresponds to the first density plot. It shows a median sale price around 200,000. Several outliers are present above approximately 500,000.The bottom-right plot corresponds to the second density plot. It exhibits fewer outliers compared to the first box plot, with a median sale price around 150,000.Both box plots have whiskers extending both ways from the boxes, suggesting overall data spread.

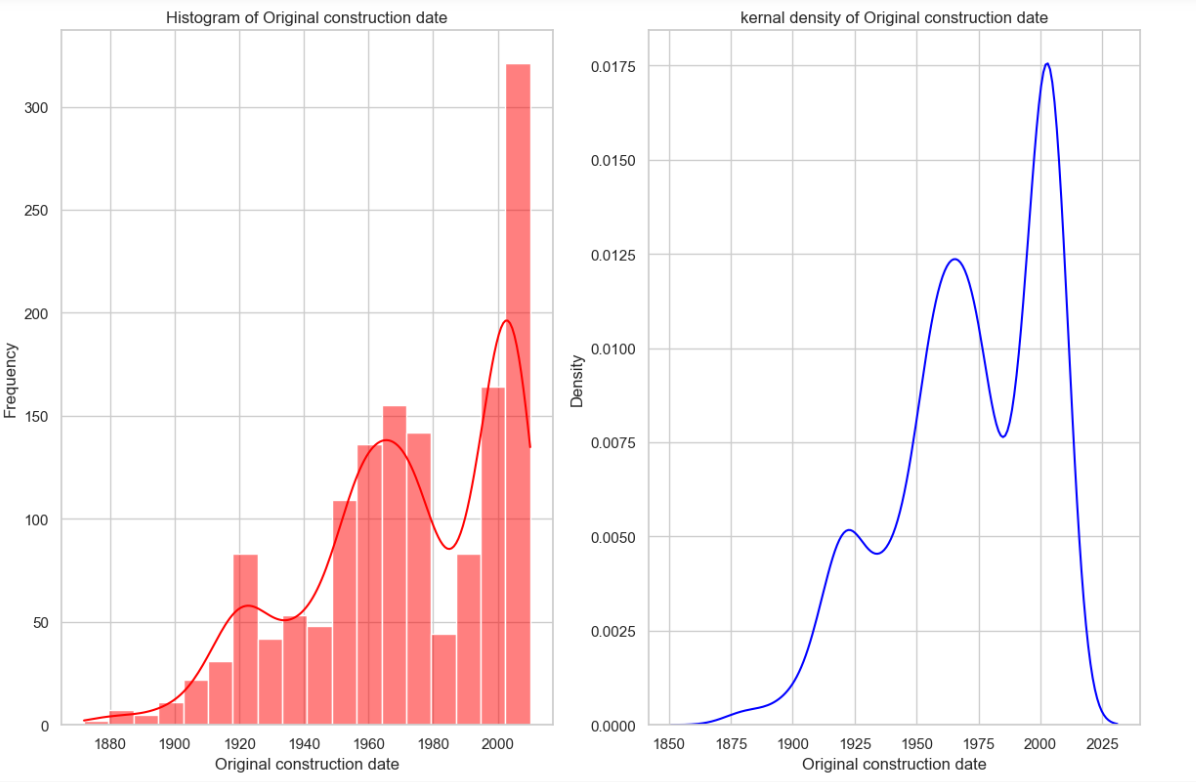
1. **Insights of Univariate Analysis:** Univariate analysis focuses on a single variable at a time. It helps us understand the distribution, characteristics, and patterns of that specific variable within a dataset.





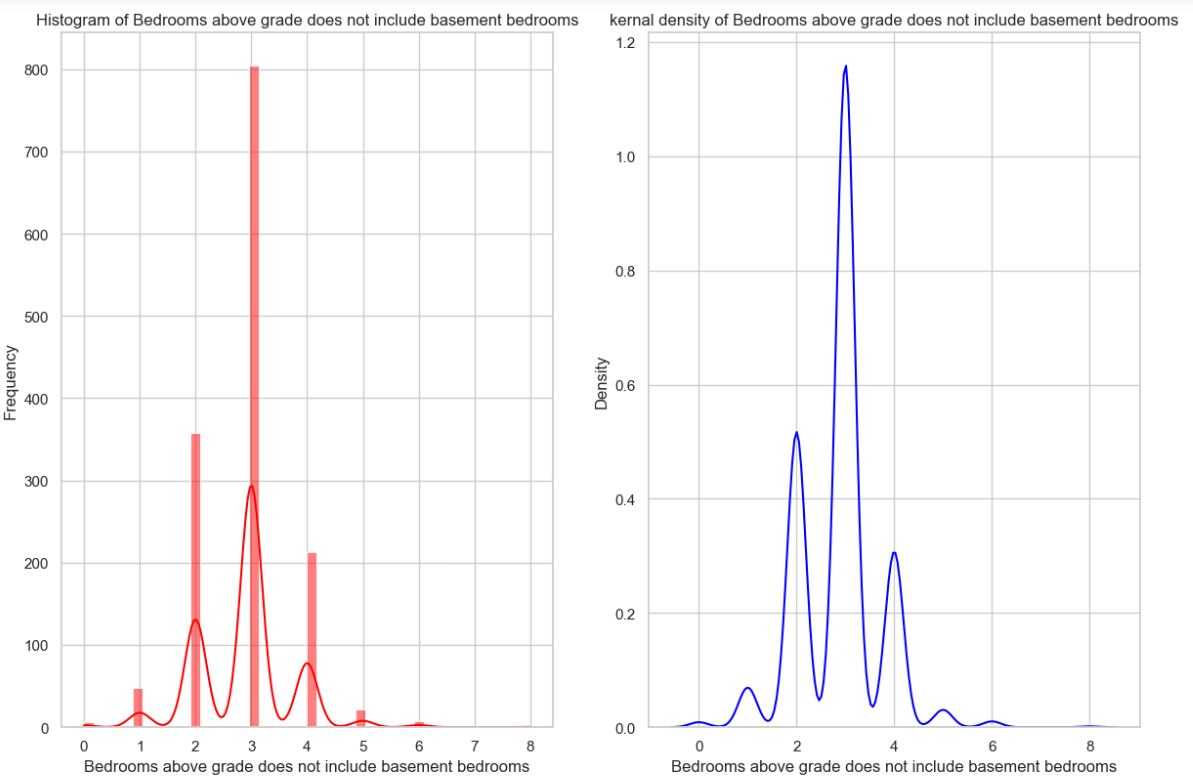
**Histogram (Hist plot) of Sale Price:**The histogram, depicted in red bars, shows the frequency of different ranges of sale prices.Most sales occur in the price range of approximately 100,000 to 200,000.There’s a rapid decline in frequency as the sale price increases.

**Kernel Density Estimate (KDE) Plot of Sale Price:**The KDE, represented by a smooth blue curve, also indicates that most sales are concentrated around the lower end, forming a peak there.Both plots suggest that the majority of sales have lower prices.



**Histogram of Original Construction Date:**The histogram on the left shows the frequency of constructions over time.The x-axis represents the original construction date, ranging from 1880 to 2020.Notable peaks occur around 1920 and just before 2000.

**Kernel Density Estimate (KDE) of Original Construction Date:**The blue line graph on the right represents the estimated probability density function.It smooths out the data from the histogram.Peaks in density align with those in the histogram, emphasizing the same construction periods.



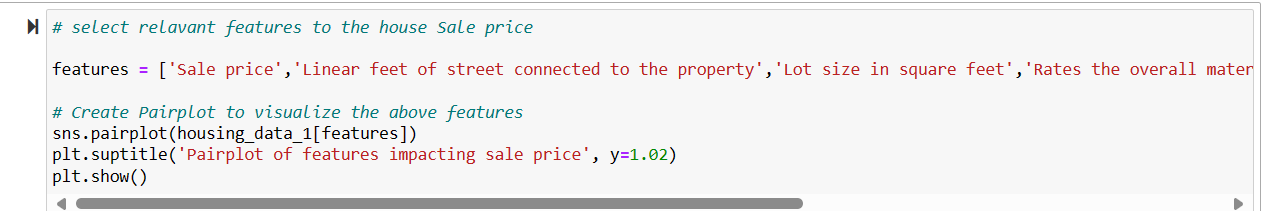
**Histogram of Bedrooms Above Grade:**The left graph is a histogram showing the frequency of homes with a given number of bedrooms above grade.The x-axis represents the number of bedrooms, ranging from 0 to 8.The y-axis shows the frequency of homes with that specific number of bedrooms.There’s a prominent peak at 3 bedrooms, indicating that many homes have this exact number of rooms above grade.

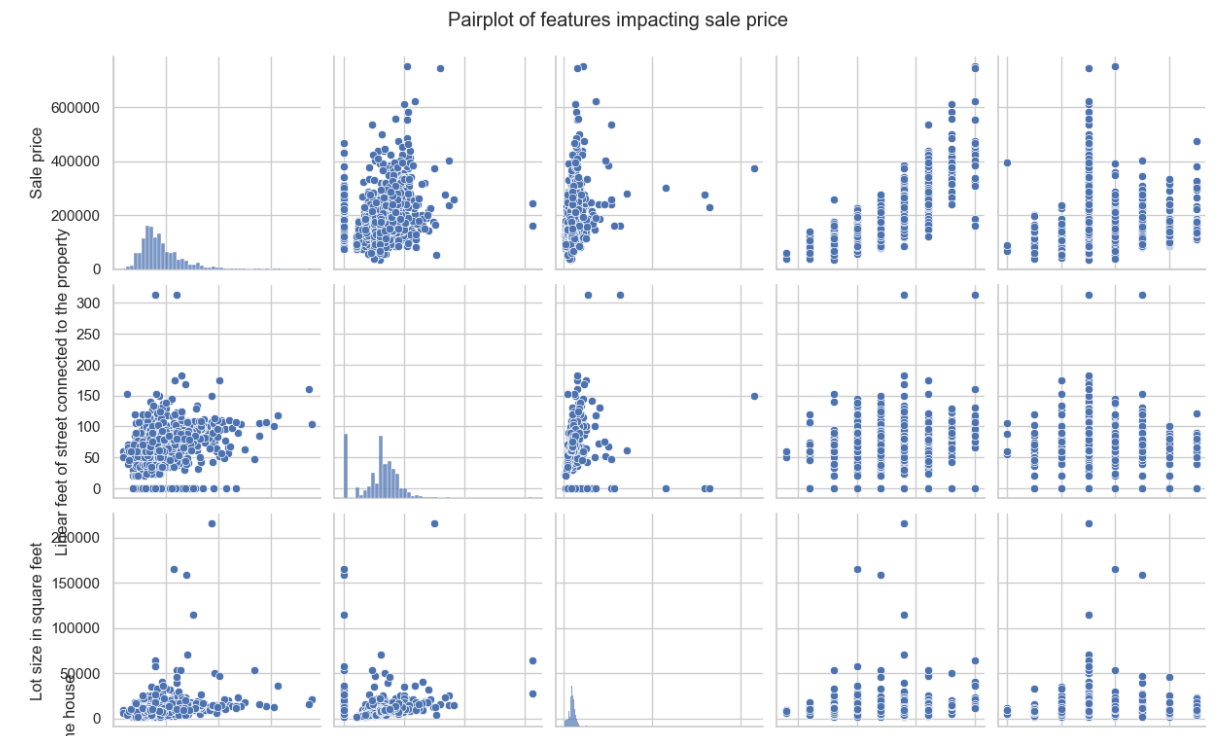
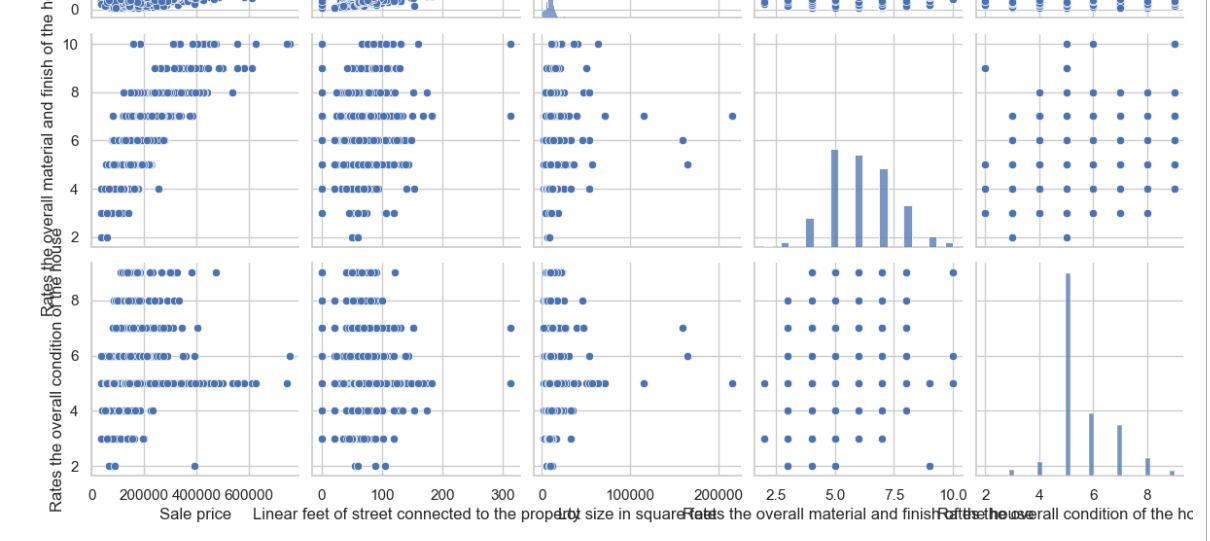
**Kernel Density Estimate:**The right graph is a kernel density estimate (a smoothed representation of the data’s distribution).It also represents the density of homes based on the number of bedrooms above grade.Similar to the histogram, there’s a peak around 3 bedrooms, suggesting that this is a common configuration for homes.

1. Insight Of Multivariate Analysis: Multivariate analysis is a powerful technique used in data analytics to explore complex datasets. Unlike univariate analysis (which focuses on a single variable) or bivariate analysis (which considers two variables), multivariate analysis simultaneously considers multiple variables. By doing so, it reveals intricate relationships, uncovers hidden patterns, and provides a deeper understanding of the underlying structure

**Advantages of Multivariate Analysis:**

The depth of insight provided by multivariate analysis is its major advantage. By exploring multiple variables, you create a more detailed picture of real-world scenarios. The insights you uncover become more applicable and actionable



  **Sale Price Distribution:** The diagonal plots show the distribution of sale prices. Most properties fall within a certain price range, but there are some outliers with significantly higher or lower prices.**Correlations:**

**Lot Area vs. Sale Price:** There seems to be a positive correlation between lot area (size of the property) and sale price. Larger lots tend to have higher prices.

**Overall Quality vs. Sale Price:** Higher overall quality ratings are associated with higher sale prices. Properties with better quality features command higher prices.

**1st Floor Square Footage vs. Sale Price:** There’s a positive correlation between the size of the first floor and sale price. Larger first floors are typically associated with higher prices.

**Interactions:** The scatter plots off the diagonal reveal how these variables interact with each other. For example, properties with larger lot areas and higher overall quality tend to have higher sale prices.

**Finding Correlations:**

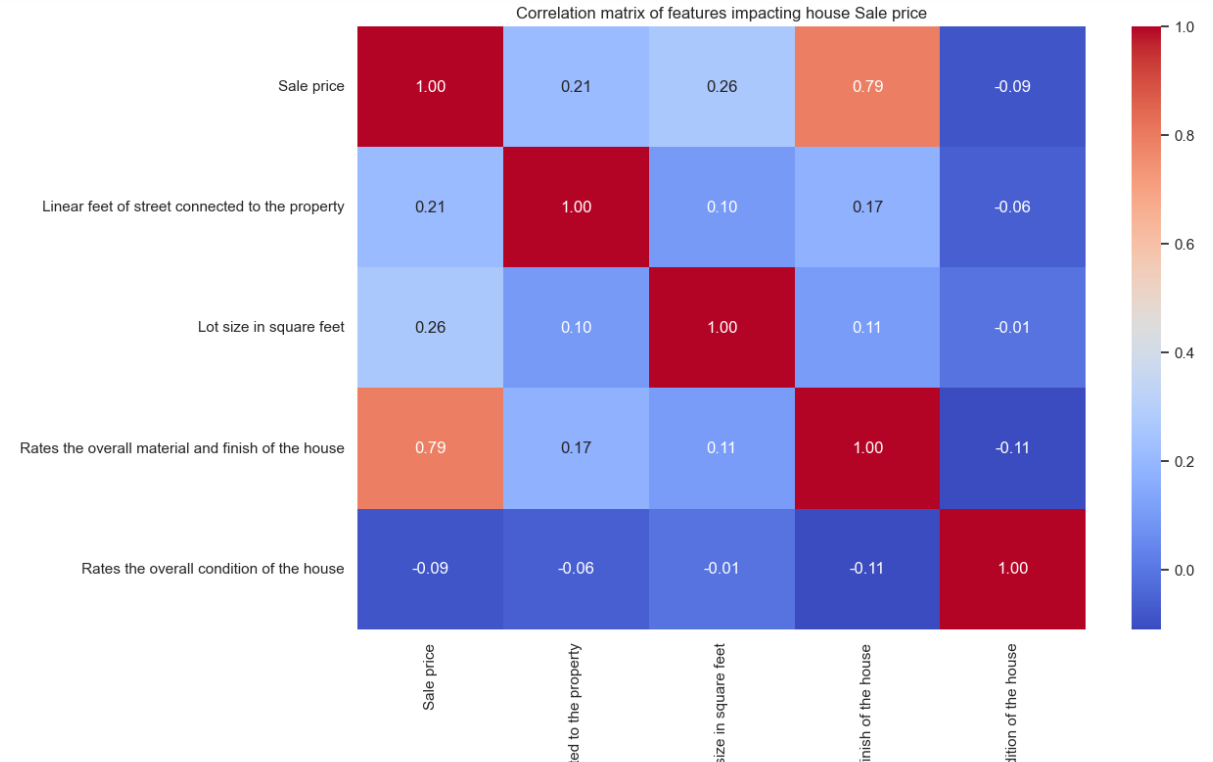


**Purpose:** The code aims to explore the relationships between different features (variables) related to house sale prices.

**Steps:Data Preparation:** It assumes that there is a dataset called housing\_data\_1 containing relevant information about houses.

**Correlation Matrix:** The code calculates the correlation matrix for specific features (columns) from the dataset. This matrix quantifies how strongly each feature is related to others.

**Heatmap:** It creates a heatmap using libraries like matplotlib and seaborn. The heatmap visually represents the correlation values. Warmer colors (reds) indicate positive correlations, while cooler colors (blues) indicate negative correlations.



**Features Included:**

**The matrix includes the following features:**Sale PriceLinear feet of street connected to the propertyLot size in square feetOverall material and finish rating of the houseOverall condition rating of the house

**Color Intensity:**The color intensity in each cell indicates the strength of correlation between the features.Darker colors represent stronger correlations.

**Positive Correlation:**There is a strong positive correlation between Sale Price and the Overall material and finish rating of the house.

**Other Correlations:**The matrix shows additional correlations between the listed features.

**Strong Positive Correlations:** Strong Positive Correlation:When the value of one variable increases, the value of the other variable also increases in a similar fashion.For instance, consider the relationship between hours studied and exam scores. As a student spends more hours studying, their exam score tends to be higher. This positive trend indicates a strong positive correlation between hours studied and exam scores.The correlation coefficient ® helps us gauge the strength of this relationship. The further away r is from zero, the stronger the correlation.

**Strong Negative Correlations:** Negative correlation is a statistical concept that describes the relationship between two variables. When these variables move in opposite directions, we say they exhibit negative correlation. Here are some key points to understand:**Definition:**Negative correlation occurs when one variable increases while the other decreases, or vice versa.It’s like a seesaw: when one side goes up, the other goes down!In contrast, positive correlation happens when both variables move in the same direction (i.e., when one increases, so does the other).