- Duplicate rows were removed.
- The 'Id' column was dropped as it is not relevant for modeling.
- Missing values in numerical columns were imputed with the median, and missing values in categorical columns were imputed with the mode.
- A new feature 'TotalBathrooms' was created by summing 'FullBath' and 'HalfBath' columns.
- The selected features for modeling were 'GrLivArea', 'BedroomAbvGr', and 'TotalBathrooms'.
- The target variable was 'SalePrice'.
- The dataset was split into training and test sets with a 80:20 ratio.
- The features in the training and test sets were standardized using the StandardScaler from scikit-learn.
- Polynomial features of degree 2 were added to the scaled features using the PolynomialFeatures transformer from scikit-learn.
- Lasso regularization was used for feature selection.
- The Lasso model was trained on the polynomial features of the training set.
- The SelectFromModel class was used to select the relevant features based on the Lasso model's coefficients.
- A Linear Regression model was initialized and trained on the selected features from the training set.
- Predictions were made on the test set.
- The model's performance was evaluated using the following metrics:
    - Mean Squared Error (MSE): 2070456205.391273
    - Root Mean Squared Error (RMSE): 45502.26593688795
    - R-squared (R2) Score: 0.6779550216322588

The key insights from the results are:

1. The Linear Regression model achieved a reasonably good R-squared score of 0.68, indicating that it can explain around 68% of the variance in the target variable (SalePrice).
2. The RMSE value of 45502.26 suggests that, on average, the model's predictions deviate from the actual sale prices by approximately 45,502. This error value should be considered in the context of the housing price range in the dataset.
3. The use of polynomial features and Lasso regularization for feature selection suggests an attempt to capture non-linear relationships between the features and the target variable, as well as to reduce the impact of multicollinearity and overfitting.
4. While the model shows promising results, there may be room for further improvement by exploring additional feature engineering techniques, trying different model types (e.g., tree-based models, ensemble methods), or fine-tuning the hyperparameters of the models.