**Importing Dependencies:**

Import necessary libraries and modules such as pandas, numpy, seaborn, matplotlib, scikit-learn (for machine learning), and xgboost.

**Loading Dataset:**

Load the housing dataset from a CSV file using the pandas library.

**Data Exploration:**

Display basic information about the dataset using dataset.info().

Generate summary statistics of the dataset using dataset.describe().

Get the column names using dataset.columns.

**Visualization and Pre-Processing of Data:**

Visualize the distribution of the target variable 'Price' using a histogram (sns.histplot) and a box plot (sns.boxplot).

Create joint plots to visualize the relationship between 'Price' and other numeric features.

Create a pair plot to visualize pairwise relationships between features.

Plot histograms of all the numeric features in the dataset.

**Visualizing Correlation:**

Calculate the correlation matrix for numeric columns in the dataset using dataset.corr().

Visualize the correlation matrix using a heatmap (sns.heatmap) with annotations.

Dividing Dataset into Features and Target Variable:

Split the dataset into features (X) and the target variable (Y).

Features include columns: 'Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms', 'Avg. Area Number of Bedrooms', and 'Area Population'.

The target variable is 'Price'.

**Using Train-Test Split:**

Split the data into training and testing sets using train\_test\_split with a 20% test size and a random seed.

**Standardizing the Data:**

Standardize the feature data using StandardScaler to ensure all features have the same scale.

Model Building and Evaluation:

**Model 1: Linear Regression (LinearRegression):**

Fit the Linear Regression model to the standardized training data.

Predict house prices on the test data and store the predictions in 'Prediction1'.

Visualize the actual vs. predicted prices using a line plot.

Calculate and print the R-squared score, mean absolute error, and mean squared error.

**Model 2: Support Vector Regressor (SVR):**

Fit the Support Vector Regressor model to the standardized training data.

Predict house prices on the test data and store the predictions in 'Prediction2'.

Visualize the actual vs. predicted prices using a line plot.

Calculate and print the R-squared score, mean absolute error, and mean squared error.

**Model 3: Lasso Regression (Lasso):**

Fit the Lasso Regression model to the standardized training data with a specified alpha value.

Predict house prices on the test data and store the predictions in 'Prediction3'.

Visualize the actual vs. predicted prices using a line plot.

Calculate and print the R-squared score, mean absolute error, and mean squared error.

**Model 4: Random Forest Regressor (RandomForestRegressor):**

Fit the Random Forest Regressor model to the standardized training data with 50 estimators.

Predict house prices on the test data and store the predictions in 'Prediction4'.

Visualize the actual vs. predicted prices using a line plot.

Calculate and print the R-squared score, mean absolute error, and mean squared error.

**Model 5: XGBoost Regressor (XGBRegressor):**

Fit the XGBoost Regressor model to the standardized training data.

Predict house prices on the test data and store the predictions in 'Prediction5'.

Visualize the actual vs. predicted prices using a line plot.

Calculate and print the R-squared score, mean absolute error, and mean squared error.