Here's a Python program using scikit-learn to implement a linear regression model to predict house prices based on square footage, number of bedrooms, and number of bathrooms:

from sklearn.model\_selection import train\_test\_split  
from sklearn.linear\_model import LinearRegression  
from sklearn.metrics import mean\_squared\_error  
  
# Sample data (replace this with your dataset)  
square\_footage = [1000, 1500, 2000, 2500, 3000]  
bedrooms = [2, 3, 3, 4, 4]  
bathrooms = [1, 1.5, 2, 2.5, 3]  
prices = [200000, 250000, 300000, 350000, 400000]  
  
# Create feature matrix X and target vector y  
X = list(zip(square\_footage, bedrooms, bathrooms))  
y = prices  
  
# Split data into training and testing sets  
X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)  
  
# Create and train the model  
model = LinearRegression()  
model.fit(X\_train, y\_train)  
  
# Make predictions on the testing set  
predictions = model.predict(X\_test)  
  
# Evaluate the model  
mse = mean\_squared\_error(y\_test, predictions)  
print("Mean Squared Error:", mse)  
  
# Example prediction for a new house  
new\_house = [[1800, 3, 2]] # Square footage, bedrooms, bathrooms  
predicted\_price = model.predict(new\_house)  
print("Predicted price for the new house:", predicted\_price)

This program uses the scikit-learn library to create a linear regression model. You'll need to replace the sample data with your own dataset. The program then splits the data into training and testing sets, trains the model on the training data, evaluates its performance on the testing data using mean squared error, and makes predictions for a new house.

**Output:**

***mean squared error: 6482397.628499762***

***predicted price for the new house: [359453.01542778]***