**Phase-2SubmissionTemplate Student Name:** [HEMALATHA P]

**RegisterNumber:**[422723104040]

**Institution:** [V.R.S. College of engineering and technology]

**Department:**[Computer Science and Engineering]

**Date of Submission:** [10.05.2025]

Topic : [Forecasting house prices accurately using smart regression techniques in data science.]

**GithubRepositoryLink:https://github.com/Hemalatha-tech-star/Hemalatha.git**

1. **Problem Statement**

The goal is to predict house prices accurately using advanced regression techniques. This is a supervised regression problem where the target variable is the price of a house. Predicting house prices accurately is vital for real estate investors, home buyers, and policy makers to make informed financial decisions.

# **2. Project Objective**

# 👉 Technical Objective: Develop and compare machine learning models to predict house prices based on various features.

# 👉Model Goals: Maximize prediction accuracy while maintaining interpretability and real-world applicability.

# 👉Update Post-EDA: After initial exploration, we identified key influencing features and the need for robust outlier handling and feature transformation.

# **3. Project Workflow**

**Start**

# 

# 

**Data Collection**

**Data Cleaning & Preprocessing**

# 

**Exploratory Data Analysis**

**Feature Engineering**

**Model Selection & Training**

**Model Evaluation**

**Visualization & Insights**

**Final Reporting**

**4. Data Description**

👉**Dataset Name** : meenakshisajan housing dataset

👉**Link: https://www.kaggle.com/datasets/meenakshisajan/housing-price-dataset**

**👉Type of Data** : Structured , tabular data

👉**Number of Records and Features:** ~2,900 records with 80 features

👉**Nature of Data:** Static

👉**Target Variable:** SalePrice (house price)

**5. Data Preprocessing**

**Link:https://colab.research.google.com/drive/1txvjJqOcY8NdbURG4Rm774doKA99F1bu?usp=sharing**

**👉 Handling Missing Values :** imputed using mean (numeric) and mode (categorical) or removed if irreparable.

**5.2 Removing or Justifying Duplicate Records**

**Duplicates:** Checked for duplicates using df.duplicated(). No exact duplicates were found, hence no records were removed.

**5.3 Detecting and Treating Outliers**

👉Outliers in features like GrLivArea and SalePrice were identified using boxplots and Z-score/Interquartile Range (IQR) methods.

👉**Extreme outliers:** houses with abnormally high GrLivArea but low SalePrice)

**5.4 Data Type Conversion and Consistency**

👉 Ensured features such as MSSubClass (nominal data represented as numbers) were converted to categorical.

👉Verified correct formats for numerical and date-like variables.

**5.5 Encoding Categorical Variables**

**👉Nominal features** (e.g., Neighborhood, Exterior1st) were encoded using One-Hot Encoding to preserve categorical nature.

**👉Ordinal features** (e.g., ExterQual, BsmtQual) were mapped using Label Encoding based on domain hierarchy (e.g., Poor < Fair < Good < Excellent).

**5.6 Feature Scaling**

👉Applied Standardization (Z-score scaling) to numerical features like LotArea, GrLivArea, and SalePrice to normalize ranges.

👉Ensured features are on a similar scale to help algorithms like Ridge and

**6. Exploratory Data Analysis (EDA)**

**👉Univariate:** Histograms and boxplots revealed skewness in SalePrice, LotArea, etc.

👉**Multivariate**: Strong positive correlation between OverallQual, GrLivArea and SalePrice.

👉I**nsights:** Quality of materials and living area are major price influencers. Some variables (e.g., PoolArea, MiscVal) have low variance and limited effect.

**7. Feature Engineering**

**👉Created Features**: TotalBathrooms, AgeAtSale, IsRemodeled, TotalSF.

👉**Transformed:** Log-transform on skewed features (SalePrice, GrLivArea).

👉**Dropped**: Low-importance or highly missing fields (e.g., Alley, Fence).

👉**Justification**: Based on correlation with target and business logic.

**8. Model Building**

**Models Used:**

👉Linear Regression (baseline)

👉Random Forest Regressor

👉XGBoost Regressor

**👉Justification:** Tree-based models handle non-linearity and interactions well.

**👉Split**: 80/20 train-test split using train\_test\_split.

👉**Metrics:** MAE, RMSE, R² Score.

**9. Visualization of Results & Model Insights**

**👉Feature Importance:** OverallQual, TotalSF, GrLivArea are top predictors.

👉**Residual Plots:** Checked for bias and homoscedasticity.

👉**Performance Plots:** Bar charts comparing RMSE and R² for each model.

👉**Conclusion**: XGBoost performed best with the lowest RMSE.

**10. Tools and Technologies Used**

**👉Language:** Python

👉**IDE:** Google Colab

👉**Libraries**: pandas, numpy, matplotlib, seaborn, scikit-learn, XGBoost

👉**Visualization**: Plotly for interactive plots

**11. Team Members and Roles**

**HEMALATHA P** - Data cleaning, EDA

**HARINI R** - Feature engineering

**HEMALATHA G -** Model development

## **HEMA G -** Documentation and reporting

# 