**HOUSE PRICE PREDICTION**

**Project Overview**

**Objective**

Predicting house prices in California based on various features such as median income, house age, and room count to help real estate businesses assess property values.

**Dataset**

Predicting house prices in California based on various features such as **median income, house age, average rooms & bedrooms, population, average occupation, latitude & longitude** to help real estate businesses assess property values.

**Tools and Technologies**

* Data Manipulation - Pandas, Numpy
* Data Visualization – Matplotlib, Seaborn
* Machine Leaning – Scikit-learn
* Model Building &Training – Linear Regression, Gradient Boost Regressor, XGBoost Regressor

### **Approach**

* **Data Preprocessing**: Preprocessed the data by handling missing values, encoding categorical variables, and splitting it into training and testing datasets.
* **Exploratory Data Analysis (EDA)**: Created pair plots and heatmaps to understand relationships between features and detect correlations, which helped in feature selection.
* **Modeling**: Used **XGBoost** for regression due to its ability to handle complex, non-linear relationships. Performed hyperparameter tuning with **GridSearchCV** to optimize parameters like learning rate, max depth, and n\_estimators.

### **Evaluation**

### **Metrics**: Evaluated model performance using **MSE (Mean Squared Error)** and **R² (R-squared)**. Model achieved a test R² of 0.85, with MSE of 0.19, showing good generalization.

**Key Findings and Insights**

* **Feature Selection**: Using features like MedInc (Median Income), HouseAge, and AveRooms helps predict housing prices accurately. Feature importance can help identify the most relevant predictors.
* **Model Tuning**: Through **GridSearchCV**, we find the best set of hyperparameters (e.g., n\_estimators, learning\_rate, max\_depth), which improves the model's performance.
* **Model Performance**: By using R² and MSE, we assess how well the model is performing. After tuning, we observe a reduction in error, indicating improved prediction quality.
* **Model Comparison**: By comparing MSE and R² scores, we can determine which model performs better. XGBoost usually provides better results due to its ability to model non-linear relationships.
* **Feature Importance**: It reveals which features are the most influential in predicting house prices. This can help in understanding the drivers of California housing markets.
* **Deployment**: Saving the model using joblib allows us to deploy the model for future predictions.

**Future Improvements:**

* Improve Feature Engineering
* Implement Deep Learning Models
* Deploy as a Web Application