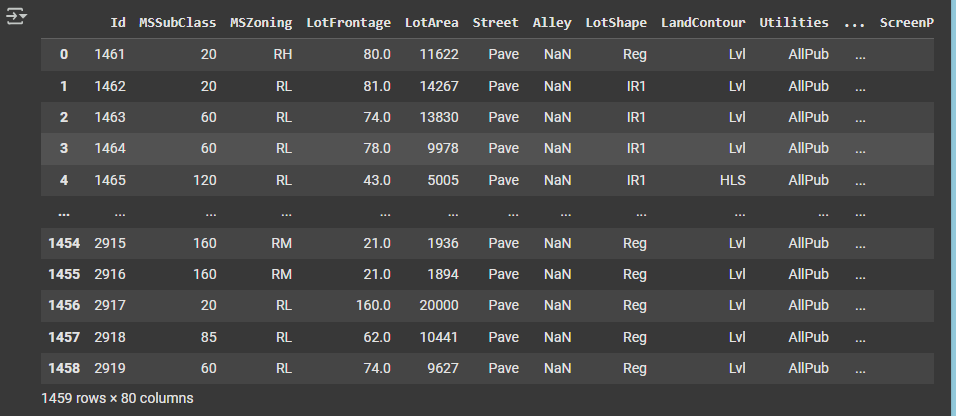
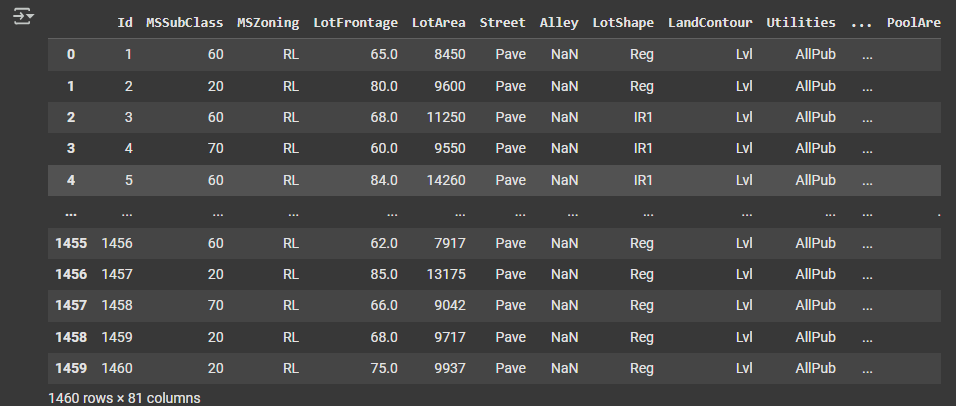
Lab task 1 (Report)

# Report of the code

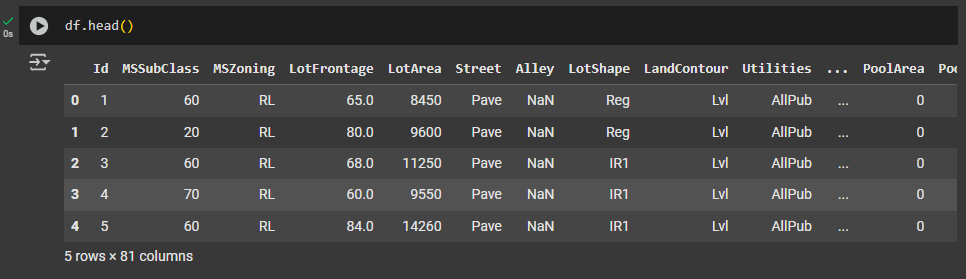
**1. Data Loading and Preprocessing:**

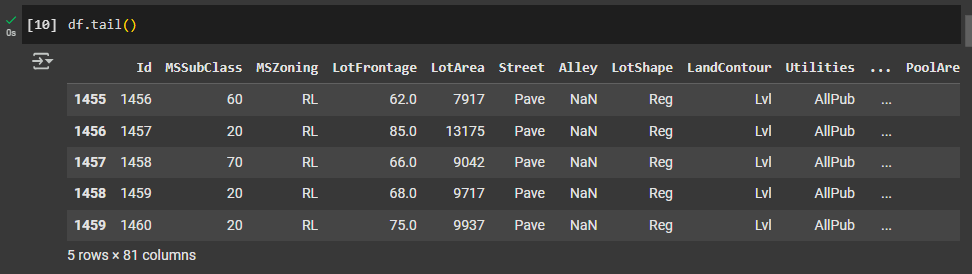
* First, importing necessary libraries like pandas and scikit-learn.
* It loads two CSV files, "train.csv" and "test.csv", into pandas DataFrames df and df2, respectively.

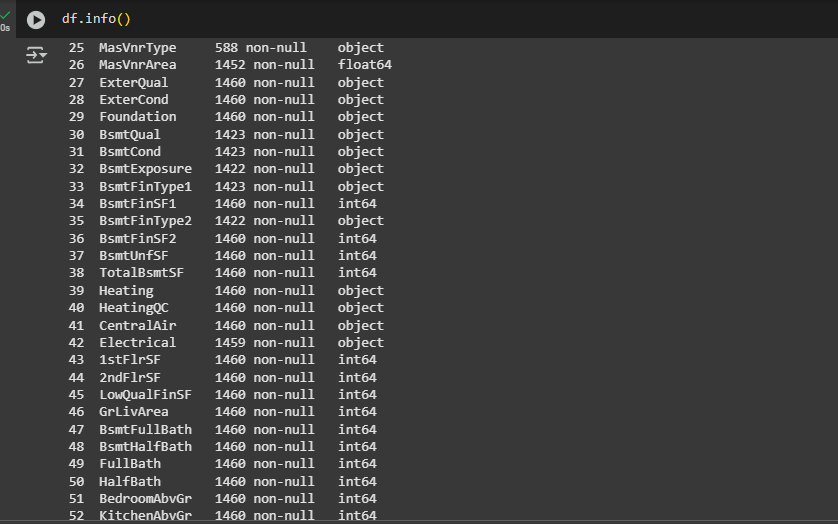


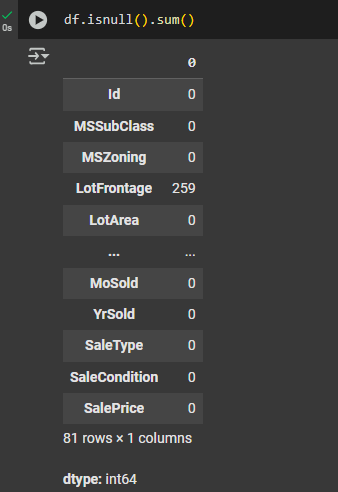


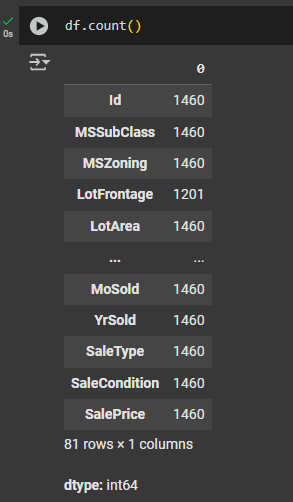
* **Data Exploration:** It then performs basic data exploration using head(), tail(), info(), isnull().sum(), and count() to understand the structure and potential issues in the data.



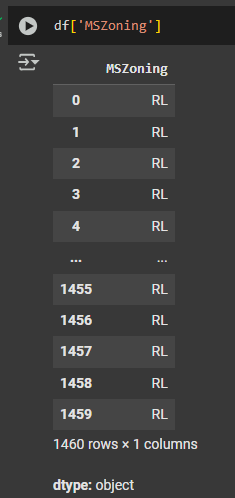


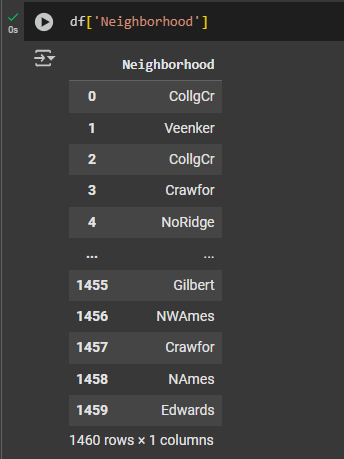






* **Handling Missing Values:** The code addresses missing values by filling them with appropriate strategies. For numerical columns like 'GarageCars', 'GarageArea', etc., it fills missing values with 0. For categorical columns like 'MSZoning', 'Neighborhood', etc., it uses the mode (most frequent value) or 'empty' as replacements.
* **Label Encoding:** To prepare categorical features for the model, it uses Label Encoding to convert them into numerical representations using sklearn.preprocessing.LabelEncoder. This is done for columns like 'MSZoning', 'Neighborhood', 'SaleCondition', 'ExterQual', 'KitchenQual', and 'FireplaceQu'.

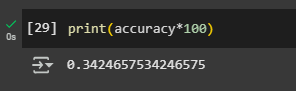






**2. Model Training and Prediction:**

* **Feature Selection:** A subset of columns is selected for training the model (stored in the cols variable). These are likely chosen based on their relevance to predicting the target variable 'SalePrice'.
* **Data Splitting:** The dataset is split into training and testing sets using train\_test\_split with a test\_size of 0.2 and random\_state set to 42 for reproducibility.
* **Model Selection:** A RandomForestClassifier is chosen for the prediction task.
* **Model Training:** The model is trained using the training data (X\_train, y\_train).
* **Prediction:** Predictions are made on the test set (X\_test) and the accuracy is calculated using accuracy\_score. The accuracy is then printed.



**3. Submission File Creation:**

* **Predictions on Test Data:** The trained model is used to predict the 'SalePrice' for the 'test.csv' data.
* **Submission DataFrame:** A DataFrame is created with 'Id' and the predicted 'SalePrice' columns.
* **Saving Submission File:** The DataFrame is saved to a CSV file named "submission.csv" and downloaded using google.colab.files.download.