Task 1-Linear Regression

1. Packages used
    - Numpy
    - Pandas
    - Sklearn
        - sklearn.linear_model
        - sklearn.model_selection
        - sklearn.metrics
    - Matplotlib
        - pyplot

2. Problem definition

   Prediction of insurance charges (output of model) for an individual based on his features (age, sex, bmi, smoker, region, and children).

3. Approach
    - Data exploration to make sure that there are neither null values nor outliers.
    - Conversion of the dataset into a numeric dataset, where the sex, smoker, and region columns become numeric values. I did this step to observe the correlation between all dataset features and select the features having the largest correlation with the output (charges).
    - Feature extraction to get the best features which are smoker, bmi and age.

   A. Implementation using sklearn
    - Implementing a model based on all features.
        - Input: all columns except charges
        - Output: charges columns
    - Split data into train (70%) and test (30%) samples.
    - Calculate the R-squared score.
    - Define inputs, targets and predictions.
    - Calculate the root mean square error.
    - Implementing a single feature linear regression model to visualize the change in R-squared score, and it decreased significantly.
    - Results:
        - R squared = 0.77
        - RMSE = 6046.98

B. Implementation using python from scratch
   - Implement ‘my_train_test_split’ function to split data into train and test datasets, it takes features data ‘X’ and target data ‘Y’ and splits them randomly.
   - Implement ‘my_linear_regression’ function that performs LR from scratch using the closed-form solution by taking features data ‘X’ and target data ‘Y’ and adding intercept, then calculates the coefficients and returns intercept and coefficients.
   - Implement ‘Predict’ function which uses the intercept and coefficients to predict the target ‘charges’ and then returns the predicted values.
   - Implement ‘my_r2’ function to calculate the r-squared score based on the equation ‘1-(ssr/sst)’
   - Calculate the r-squared score.
   - Calculate RMSE using the same equation as the implemented before.
   - Results:
     - R squared = 0.77
     - RMSE = 6046.98

4. Conclusion
   - Both implementation approaches showed the same results as I followed the same algorithm of scikit learn.
   - I tried making the input as the 3 best features (smoker, age, and bmi) and it showed a minimal difference in the rmse score (around 0.001%) and this would be a better choice than selecting all features as it reduced the model complexity into the half with approximately same model accuracy, but I didn’t include this trial in the notebook.