**Task-1**

Generate a dataset for simple linear regression problem (dataset with one input and one output column) using numpy. Add some noise also to avoid perfect or absolute linearity. The dataset must be almost linear (as discussed in the class).

Analyze the data distribution, i.e. check the linearity in data points using 2D plots.

Use inbuilt linearregression method to get the values of coefficients (m and c).

Now implement the simple linear regression from scratch using Gradient Descent algorithm, to find the optimized values of **m** and **c**. Divide the dataset into train and test sets and compute m and c using train set. Run this algorithm two times, one with lower values of m and c and another with higher values of m and c. Now compare the outcomes.

Compare the values of m and c resulted by inbuilt method and the Gradient Descent method.

Now, compute the output for test data using your model and compute the Mean Absolute Error, Root Mean Squared Error, R2 score.

Plot the linear regression model (line) on the training set and test set to represent the model fitting on the data points.

Plot the line fitted on train set after each epoch to represent the update in the line’s position based on each update in m and c. For example, if 30 epochs are run, 30 lines must be plotted on the training data points in a single plot.

Make a 3D plot to represent the change in loss with respect to changes in m and c values upon each update starting from randomly initialized values to the optimized values.

Make a 2D plot between epochs and the loss to show the decrease in loss function.