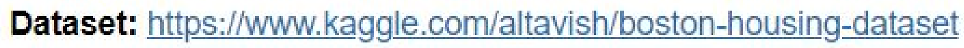
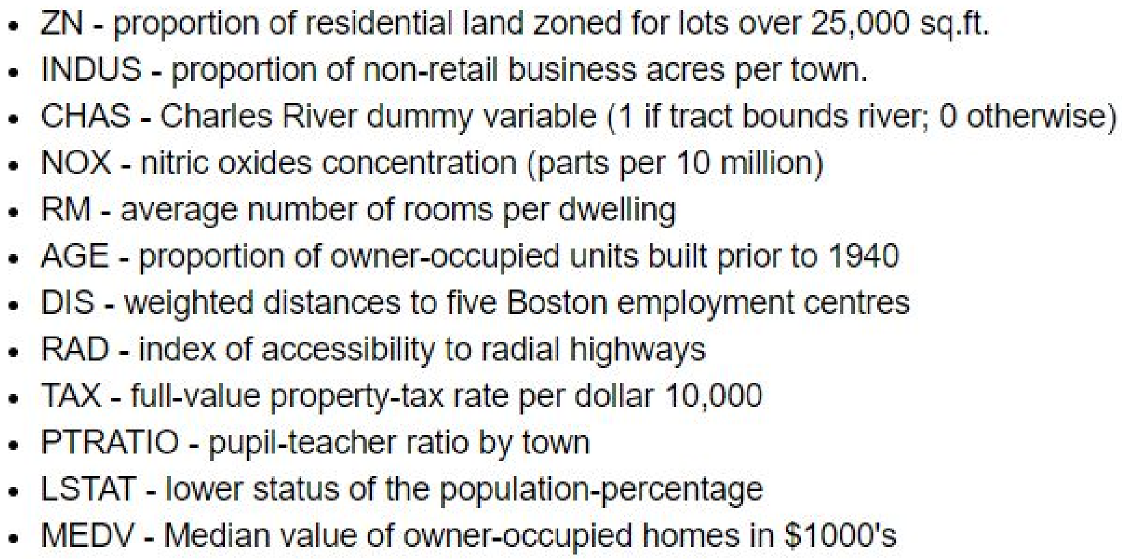
Challenge: Create a Jupyter notebook doing the following:

1. Install libraries and upload the housing data



Dataset: https://www.kaggle.com/altavish/boston-housing-dataset



- ZN - proportion of residential land zoned for lots over 25,000 sq.ft.
- INDUS - proportion of non-retail business acres per town.
- CHAS - Charles River dummy variable (1 if tract bounds river; 0 otherwise)
- NOX - nitric oxides concentration (parts per 10 million)
- RM - average number of rooms per dwelling
- AGE - proportion of owner-occupied units built prior to 1940
- DIS - weighted distances to five Boston employment centres
- RAD - index of accessibility to radial highways
- TAX - full-value property-tax rate per dollar 10,000
- PTRATIO - pupil-teacher ratio by town
- LSTAT - lower status of the population-percentage
- MEDV - Median value of owner-occupied homes in $1000's

2. Get the shape of it
3. Drop the columns named ‘CRIM’ and ‘B’
4. Check how many are null per column
5. Drop all the na’s
6. Create a new column called ‘above median’ and put a 1 if it is above the median house price and a 0 if it is below the median house price. Use the function np.where.
7. Create a dataframe called house_data_selected that consists of the MEDV, RM, DIS, and AGE columns. Show the head of your dataframe
8. Show all the relevant statistics for your data.
9. Create a scatterplot of the AGE vs MEDV and another of the RM vs MEDV.
10. Use pairplot to analyze your 4 four variables
11. Create a correlation matrix
12. Use the seaborn lmplot to do a linear regression with RM as the x, and MEDV as the y. Name the chart ‘Salary’
13. Use stats.from scipy to use the stats.linregress function with x as RM and y as MEDV to find the slope, intercept, and r_value.
14. Analyze AGE vs MEDV.
15. Set x to everything but MEDV (using the drop function), and set y to MEDV, and import statsmodels.api as sm, then use sm.OLS(y,X).fit() and output the summary() of the model.