Python Machine Learning for Dummies: Scikit-Learn Tutorial for Beginners:
Clean our data which in our case is titanic before we start filling in all the empty data columns with Nan and drop columns in the data that are not relevant to whatever we are testing, which in this case is the survival rate of people on the titanic so the data from such as cabin isn't important so we drop that. Afterwards, we convert word data into numbers because the machine cannot interpret words, only integers so we convert the data types into 1 and 0s to represent “survive” and “not survive” and that will be interchanged depending on the results of our simulation. Then, we created a testing and training set, kinda like flashcards where on the x, we give 75% of our data of flashcards for the machine to learn, and then keep the 25%. 75% of the flashcards or x data type answers are shown to the model, and then the model will predict the results of the survival chance based on the provided data. Then, we get into the testing phase where we use the 25% that allows the model to see only the “front” of the flashcard and then make predictions, and then we compare to see how many predictions our model got right. This is where the built-in KNN and "metric" : ["euclidean", "manhattan", "minkowski"] comes into play and we choose the one that performs the best. Finally, we plotted to use the confusion matrix and an accuracy calculation to determine how accurate our model is to our tests, if it was a good or bad model. Lastly, we make a visual representation of the data by plotting it using built in libraries so we can see the data and information visually.

Machine Learning Pipelines in Python: Step-by-Step Guide with Scikit-Learn
For this video, we are trying to use a machine learning pipeline to predict Co2 emissions, and it starts once again, by cleaning the data. It is pretty similar to the first video where we fill in the missing values, but this time, we separate the data columns between categorical and numerical. For this Machine Learning model, they used Random Forest model and split the data between 80% training data and 20% testing datasets, fits the pipeline into the training data to make predictions on testing data. Afterwords, calculations were made for R^2, RNSE, and MAEto see how well the model performed before creating a visual bar chart for this data for comparing our calculations and also create a scatter plot to verify actual and prediction results of CO2 edmissions