Project Goal

This project aims to compare the performance of various supervised learning methods on a binary classification problem, which will help you understand each classification algorithm's advantages and disadvantages.
Project Introduction
Write python code to compare the performance of three different classification methods. You can have to use all three classification methods:

- Decision Tree
- Random Forest
- KNN

You can refer to any python libraries (pandas, numpy, matplotlib, seaborn, scikit-learn, ...) to implement the classification methods. However, your code must include the following steps:

1. Indicate the imported packages/libraries
2. Load the dataset and print the data information
3. Understand the dataset
    1. Print out the number of samples for each class in the dataset
    2. Plot some figures to visualize the dataset (e.g., histogram, etc.)
    3. For each class, print out the statistical description of features (e.g., the input variable x), such as mean, std, max and min values, etc.
4. Split data into a training dataset and a testing dataset (i.e., 80% v.s. 20%)
5. For each classification algorithm you chose, please complete the below steps in Python:
    1. Train the model using the training dataset.
        1. If there are hyperparameters in the algorithm, please use K-Fold Cross Validation (e.g., you could choose k = 5 for K-Fold Cross Validation) to tune the hyperparameters of the algorithm (e.g., explore the best value for hyperparameter “k” for KNN, or the best kernel for kernel SVM, etc.).
        2. Please use different evaluation metrics, including precision, recall, accuracy, and F1-Score, to pick up a model that gives you the best result on the validation dataset (e.g., via the Cross Validation, for kNN model, which k value gives the best precision, recall, accuracy, and F1-Score respectively)
    2. Test the model (the best one you obtained from the above stage) on the testing dataset
        1. Plot the confusion matrix
        2. Please use different evaluation metrics, including precision, recall, accuracy, and F1-Score, to report the performance of the algorithm, you can use tables or plot figures to summarize the results