1. Python (without Numpy) ⇒ [Module_1, Module_2(upto 12.37)]
    a. Matrix Multiplication
    b. Proportional Sampling
    c. Replace numbers with #,
        i. Ex 1: A = 234 Output: ###
        ii. Ex 2: A = a2b3c4 Output: ###
        iii. Ex 3: A = abc Output: (empty string)
        iv. Ex 5: A = #2a$#b%c%561# Output: ####
    d. Print name of students
        i. who got top 5 ranks
        ii. Who got least 5 ranks
        iii. Who got marks in IQR
    e. Print 5 closest elements for a given point (p,q) based on the angle between (p,q) and (x,y)
    f. from the given set of hyperplanes find a hyperplane that will separate both red and green points
    g. Given two columns of data, the first column F will have 5 unique values and the second column S will have 3 unique values (0,1,2).
        i. Find P(F_1|S==0), P(F_1|S==1), P(F_1|S==2)
        ii. Find P(F_2|S==0), P(F_2|S==1), P(F_2|S==2)
        iii. Find P(F_3|S==0), P(F_3|S==1), P(F_3|S==2)
        iv. Find P(F_4|S==0), P(F_4|S==1), P(F_4|S==2)
        v. Find P(F_5|S==0), P(F_5|S==1), P(F_5|S==2)
    h. Filling the missing values in the specified format
        i. _,_,_,value ex: _,_,_,40 ⇒ 10,10,10,10
        ii. value, _,_,_,value ex:60,_,_,_,40 ⇒ 20,20,20,20,20
        iii. Value, _,_,_ ex: 40, _,_,_ ⇒ 10,10,10,10
    i. Given two sentences S1, S2
        i. Number of common words between S1, S2
        ii. Words in S1 but not in S2
        iii. Words in S2 but not in S1
    j. Given two columns of data Y, Y_Score calculate the value of this function f(Y,Y_score) = -1*\frac{1}{n}\Sigma(Y Log10(Y_score)+(1-Y)log10(1-Y_score))
2. EDA ⇒ [Module_1, Module_2(upto 12.37)]
    a. Get insights from data, given set of questions we need to answer them by doing a bit of analysis
    b. Need two data sets of similar type one for reference, one for assignment
3. Implementing TFIDF vectorizer ⇒ [Module_1, Module_2, Module_3(upto 18.14)]
    a. Build a TFIDF Vectorizer, given the reference for countvectorizer
    b. Implement min_df and max_feautres attributes

4. Implement RandomSearchCV with k fold cross validation on KNN ⇒ [Module_1, Module_2, Module_3(upto 19.31)]
    a. For each hyper parameter select two disjoint set of indices and divide the data into train and test
    b. Train model on train data and find the performance metric value on test data
    c. Calculate the average performance metric score for each hyper parameter

5. Compute Performance metrics without Sklearn ⇒ [Module_1, Module_2, Module_3(upto 22.8)]

    Given original and predicted values (without sklearn)

    a. #P >> #N
        i. Calculate F1 Score
        ii. Calculate AUC
        iii. Calculate Accuracy
        iv. Calculate confusion matrix
    b. #P << #N
        i. Calculate F1 Score
        ii. Calculate AUC
        iii. Calculate Accuracy
        iv. Calculate confusion matrix
    c. Find out the best threshold from given probability scores that will give the lowest score f(y,y^predict) = a*fpr+b*fnr, here y^pred = [0 if y_score < y_threshold else 1]
    d. Given y and y_predicted(both are real-valued features)
        i. Calculate MSE
        ii. MAPE
        iii. R Squared Error
6. Apply Multinomial NB on Donors Choose Dataset ⇒ [Module_1, Module_2, Module_3(upto 24.20)]
7. Implement SGD Classifier with Log Loss and L2 regularization Using SGD: without using sklearn ⇒ [Module_1, Module_2, Module_3(upto 27.11)]

8. How each model behaves ⇒ [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4(upto 29.14)]
    a. Given a highly imbalanced dataset (we will be incrementing imbalance in data adding few data points) observe how each model behaves
        i. Draw hyperplane in Logistic regression
        ii. Draw hyperplane in SVM
        iii. Draw decision boundary in KNN
    b. Given data with a set of outliers (we will be adding outliers in data incrementally)
        i. How hyperplane changes in Logistic Regression
        ii. How hyperplane changes in SVM
    c. elliptical data with one or two outliers linear regression
    d. Given 3d data points, such a way that var(3)>>(var2)>var(1)
        i. How hyperplane changes before and after standardization of data in Logistic Regression
        ii. How hyperplane changes before and after normalization of data in Logistic Regression
        iii. How hyperplane changes before and after standardization of data in SVM
        iv. How hyperplane changes before and after normalization of data in SVM
    e. Create a dataset with features [X, X^2, 2*X, Y, Z, X+Y] and perform perturbation test (iris.csv)
    f. Visualization of the weight vector D1(X1, X2) with and without regularization D2(X1, X2, X3) with and without regularization
    g. How the distribution of weights changes with increasing of lambda(L2) values for a dataset with 1000 features

9. Apply Decision Trees on Donors Choose Dataset: [Module_1, Module_2, Module_3, Module_4(upto 31.14)]
10. Application of Bootstrap samples in Random Forest: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4(upto 33.8)]
    a. Choose any base model of your choice(either a Decision tree or Logistic regression). You can choose 30 to 40 base models based on your RAM.
    b. Do both Row and column sampling to train each of the base learners.
    c. Find the confidence interval on AUC based on the results of base learners
11. Apply GBDT/XGBOOST/LIGHT-GBM on Donors Choose Dataset: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4(upto 33.18)]
12. Clustering on Graph Dataset: [Module_1, Module_2, Module_3, Module_4, Module_7(upto 45.9)]

13. Recommendation Systems and Truncated SVD: Implement SGD algorithm to predict the ratings that user is going to give to given movie.
    Provided reference notebook [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_7(upto 46.14) and case study 9]
14. Microsoft Malware detection Case Study assignment: [Module_1, Module_2, Module_3, Module_4, Module_5, Module_6(refer case study 6)]
15. Facebook Assignment: [Module_1, Module_2, Module_3, Module_4, Module_5, Module_6(refer case study 3)]
16. SQL Assignment: [Module_1(upto 9.27)]
17. Implement a backpropagation on a given computation graph: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 50.14), https://www.youtube.com/watch?v=i94OvYb6noo, we will be providing reference videos and notebooks]
    a. Reference will be given for a couple of computational graphs
18. Tensorflow Assignment, working with callbacks and vanishing gradient problem: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 50.14), we will be providing reference videos and notebooks]
19. Given an rvl-cdip dataset, classify the given document using transfer learning: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 53.18), we will be providing reference videos and notebooks]
    a. Model 1: **INPUT --> VGG-16 without Top layers(FC) --> Conv Layer --> Max pool Layer --> 2 FC layers --> Output Layer** Train only new Conv block, FC layers, output layer. Don't train the VGG-16 network.
    b. Model 2: **INPUT --> VGG-16 without Top layers(FC) --> 2 Conv Layers identical to FC --> Output Layer** Train only last 2 Conv layers identical to FC layers, 1 output layer. Don't train the VGG-16 network.
    c. Model 3: '**INPUT --> VGG-16 without Top layers(FC) --> 2 Conv Layers identical to FC --> Output Layer**' and train only Last 6 Layers of VGG-16 network, 2 Conv layers identical to FC layers, 1 output layer.
20. Classifying CIFAR-10 dataset images with DenseNet and work with optimization: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 53.18), we will be providing reference videos and notebooks]
    a. Reference will be given for Dense-net architectures
21. Object detection - YOLO pretrained model on image net dataset: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 53.18), we will be providing reference videos and notebooks]
22. CNN with text dataset: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 53.18), we will be providing reference videos and notebooks]
23. LSTM with Text and categorical data: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 54.10), we will be providing reference videos and notebooks]

    a. Model 1: Glove embedding on text data, embedding layers on categorical features, dense layers for numerical features
    b. Model 2: Glove embedding on text data(consider words with TF IDF values within IQR), embedding layers on categorical features, dense layers for numerical features
    c. Model 3: Glove embedding on text data → LSTM, one hot encode the categorical and merge all of them → CNN1D. Merge both the outputs of LSTM and CNN1D
24. LSTM with Time series data: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 54.10), we will be providing reference videos and notebooks]
25. Encode-decoder Architecture for text abstraction, seq-seq: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 54.10), we will be providing reference videos and notebooks]
26. Siamese Architecture for Q and A selection: [We will be updating assignments soon, Module_1, Module_2, Module_3, Module_4, Module_5, Module_8(upto 54.10), we will be providing reference videos and notebooks]
27. Personal Case study -1: ML/RS
28. Personal Case study -2: DL
29. Blog on Personal Case study
30. Blog on given concept