PRACTICE TUTORIAL

- Download “Folio Leaf” dataset at link: https://drive.google.com/drive/folders/1iE9NMQ9Rlgj7EqOanL3mYmERy0v7IGA3?usp=sharing

- Upload dataset on Google Drive.

- Do practice following the steps below:

<u>Step 1</u>*: *Creating set X and set y (original)

+ Initialize empty list X and empty list y (X = [ ], y = [ ])

+ Use the **<u>os</u>** library to read each folder (import os)

+ At each folder, using **<u>cv2</u>** library to load each image as grayscale image

<u>For example:</u>

>> import cv2

>> matrix_img = cv2.imread(‘path_image’, 0)

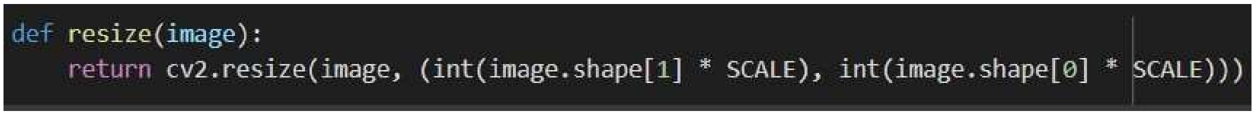
where:

- path_image: path to the folder storing the image
- matrix_img: the matrix 2D of image.

+ For each image, resize its height and width with the scale equals to 0.1 (e.g., the current width is 100. After adjusting, the width is 10) (using **cv2** library).

<u>Note:</u>

In computer, when mentioning to a 100 x 300 image, that means the width is 100 and the height is 300. But when loading this image by using cv2 library (cv2.imread), the width and height are reversed (i.e., the width is 300, the height is 100). To keep the width and height at the original proportions, when using cv2 to resize the image, we do the following:



```
def resize(image):
    return cv2.resize(image, (int(image.shape[1] * SCALE), int(image.shape[0] * SCALE)))
```

+ Add the matrix of resized image to list X, add the name of folder are being read to list y.

+ Convert the data type of X and y to np.array (e.g., X = np.array(X)).

If doing correctly steps above, the result of command “print(X.shape)” will be “(199, 232, 412)” (if do not keep the width and height at the original proportions) or “(199, 232, 412)” (if keep the width and height at the original proportions).

***Step 2:* Splitting X and y to *training set* và *testing set* in 7:3 ratio and random seed = 1.**

***Step 3:* Creating new two training sets by using 2 methods in Practice_3.pdf**

The first method - page 17

The second method - page 20

Note: For the second method, the values of hyperparameters are:

+ orientations = 9

+ pixels_per_cell = (16, 16)

+ cells_per_block = (2, 2)

+ block_norm = ‘L2’

***Step 4*: Initializing two Logistic Regression models with the hyperparameters value:**

+ max_iter = 10000

+ random_state = 1

+ n_jobs = 2

<u>Step 5:</u> Performing model <u>training</u> and <u>testing</u>.

The metrics are used for evaluating the performance's model:

+ accuracy

+ precision

+ recall

+ F1 score