Implementing a Multi-Class Classification Model using Deep neural network

In this exercise, you will build a deep neural network (DNN) model to recognize different types of flowers. The dataset contains images of various flowers, and the task is to classify these images into multiple categories. The dataset contains five types of flowers. Daisy, dandelion, rose, sunflower and tulip.


```
Visualizing unique labels in the testing dataset...
Unique labels: [0 1 2 3 4]
```

Label 0 --- > daisy

Label 1 --- > dandelion

Label 2 --- >roses

Label 3 --- >sunflowers

Label 4 --- >tulips

**Instructions: **

- You will implement a deep neural network model, which handle multiple classes.

- The goal is to classify each image into one of the predefined flower categories.

**You will learn to: **

- Build the Deep neural network (L-Layers) model from scratch.

- Implement the SoftMax activation function and compute the cost function.

- Optimize the parameters using gradient descent.

- Make predictions using the trained model.

Problem Statement

You are given a dataset (`Tr.h5`) containing:

- A training set of `m_train` images with labels indicating different flower types.

- `Te.h5` contains set of images with corresponding labels.

- Each image is of shape (num_px, num_px, 3), where 3 represents the RGB channels.

Your task is to:

1. Build a Deep neural network (L-layers) model to classify the images into different flower categories.

2. Implement forward and backward propagation to optimize the model parameters.

3. Evaluate the model's performance on the test set.

**Exercise: **

1. Load and preprocess the data.

2. Implement the softmax activation function.

3. Define the cost function for multi-class classification.

4. Implement the gradient descent algorithm to update the model parameters.

5. Test the model on the test set and evaluate its performance.

Let's start by loading and inspecting the dataset.