

```

import numpy as np

import matplotlib.pyplot as plt

import cv2

import os

import tensorflow as tf

from keras.models import load_model

from tkinter import *

import tkinter.messagebox

import PIL.Image

import PIL.ImageTk

from tkinter import filedialog

from tkinter import filedialog

import csv

file = open('solution.csv')

type(file)

csvreader = csv.reader(file)

header = []

header = next(csvreader)

CATEGORIES = ["Apple___Black_rot",
"Apple___healthy","Corn_(maize)___healthy","Corn_(maize)___Northern_Leaf_Blight","Peach___B
acterial_spot","Peach___healthy","Pepper,_bell___Bacterial_spot","Pepper,_bell___healthy","Potat
o___Early_blight","Potato___healthy","Potato___Late_blight","Tomato___Bacterial_spot","Tomato
___Late_blight","Tomato___Leaf_Mold"]

root = Tk()

root.title("Fertilizer Detection System")

root.state('zoomed')

root.configure(bg='#D3D3D3')

root.resizable(width = True, height = True)

```

```
value = StringVar()

panel = Label(root)

model = tf.keras.models.load_model("CNN.model")


# import the opencv library

import cv2


def Camera():

    # define a video capture object

    vid = cv2.VideoCapture(0, cv2.CAP_DSHOW)


    while(True):


        # Capture the video frame

        # by frame

        ret, frame = vid.read()


        # Display the resulting frame

        cv2.imshow('frame', frame)

        cv2.imwrite(r'C:\Users\divya\OneDrive\Desktop\JIT-FERTILIZER\main.jpg', frame)


        # the 'q' button is set as the

        # quitting button you may use any

        # desired button of your choice

        if cv2.waitKey(1) & 0xFF == ord('q'):

            break


    # After the loop release the cap object

    vid.release()

    # Destroy all the windows

    cv2.destroyAllWindows()
```

```

def prepare(file):
    IMG_SIZE = 150
    img_array = cv2.imread(file,1)
    #img_array=cv2.equalizeHist(img_array)
    #ret,img_array = cv2.threshold(img_array,170,155,cv2.THRESH_BINARY)
    #img_array = cv2.Canny(img_array, threshold1=50, threshold2=10)
    #img_array = cv2.medianBlur(img_array,1)
    #cv2.imshow("hello",img_array)
    new_array = cv2.resize(img_array, (IMG_SIZE, IMG_SIZE))
    new_array=np.expand_dims(new_array, axis=0)
    return new_array

```

```

def detect(filename):
    prediction = model.predict(prepare(filename))
    prediction = list(prediction[0])
    print(prediction)
    l=CATEGORIES[prediction.index(max(prediction))]
    print(CATEGORIES[prediction.index(max(prediction))])
    value.set(CATEGORIES[prediction.index(max(prediction))])
    i=int(prediction.index(max(prediction)))
    j=0
    import csv
    file = open('solution.csv')
    type(file)
    csvreader = csv.reader(file)
    header = []
    header = next(csvreader)
    for row in csvreader:
        if j == int(prediction.index(max(prediction))):
            x=header[0]+" : "+row[0]

```

```
tkinter.messagebox.showinfo("",x)

j=j+1
```

```
def ClickAction(event=None):
```

```
    filename = filedialog.askopenfilename()

    img = PIL.Image.open(filename)

    img = img.resize((250,250))

    img = PIL.ImageTk.PhotoImage(img)

    global panel

    panel = Label(root, image = img)

    panel.image = img

    panel = panel.place(relx=0.435,rely=0.3)

    detect(filename)
```

```
button = Button(root, text='ACTIVATE CAMERA', font=(None, 18), activeforeground='red', bd=20,
bg='cyan', relief=RAISED, height=3, width=20, command=Camera)
```

```
button = button.place(relx=0, rely=0.05)
```

```
button = Button(root, text='CHOOSE FILE', font=(None, 18), activeforeground='red', bd=20,
bg='cyan', relief=RAISED, height=3, width=20, command=ClickAction)
```

```
button = button.place(relx=0.40, rely=0.05)
```

```
result = Label(root, textvariable=value, font=(None, 20))
```

```
result = result.place(relx=0.465,rely=0.7)
```

```
root.mainloop()
```